



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

**“SISTEMA DE CONTROL BIOMÉTRICO PARA LOS EMPLEADOS
DE LA “FUNDACIÓN AMIGA” DE LA CIUDAD DE ESMERALDAS,
APLICANDO LA NORMA ISO/IEC 9126-2 Y 9126-3 PARA
DETERMINAR LA EFICIENCIA DEL SOFTWARE”**

Trabajo de titulación para optar al grado académico de:

INGENIERA EN SISTEMAS INFORMÁTICOS

AUTORA: ERIKA PAOLA REINA GUAÑA

TUTORA: ING. GLORIA ARCOS

Riobamba-Ecuador

2017

©2017, Erika Paola Reina Guña

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor. 0988140496

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

El Tribunal del Trabajo de Titulación certifica que: El Trabajo de investigación: “SISTEMA DE CONTROL BIOMÉTRICO PARA LOS EMPLEADOS DE LA “FUNDACIÓN AMIGA” DE LA CIUDAD DE ESMERALDAS, APLICANDO LA NORMA ISO/IEC 9126-2 Y 9126-3 PARA DETERMINAR LA EFICIENCIA DEL SOFTWARE”, de responsabilidad de la señorita: Erika Paola Reina Guaña, ha sido minuciosamente revisado por los miembros del Tribunal, quedando autorizada su presentación.

NOMBRES	FIRMAS	FECHA
Ing. Washington Luna	_____	_____
DECANO DE LA FACULTAD		
DE INFORMÁTICA Y ELECTRÓNICA		
Ing. Patricio Moreno	_____	_____
DIRECTOR DE LA ESCUELA		
DE INGENIERÍA EN SISTEMAS		
Ing. Gloria Arcos	_____	_____
DIRECTOR DE TESIS		
Ing. Eduardo Villa	_____	_____
MIEMBRO DEL TRIBUNAL		

Yo, Erika Paola Reina Guña soy responsable de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación y el patrimonio intelectual del Trabajo de Titulación pertenece a la Escuela Superior Politécnica de Chimborazo.

Erika Paola Reina Guña

DEDICATORIA

El presente trabajo de titulación está dedicado primeramente a Dios por ser mi fortaleza diaria, a mis padres Lidia Guaña y Jhonny Reina por brindarme su apoyo incondicional en mi carrera universitaria y de manera especial a mi abuelita Zoila Quingaluisa por siempre darme el aliento para alcanzar mis metas.

Erika

AGRADECIMIENTO

Agradezco a Dios y a mi familia que con apoyo, comprensión y amor me dieron las fuerzas para seguir adelante y superar los obstáculos que se me presentaron al alcanzar mi meta profesional.

A la Ing. Gloria Arcos, al Ing. Eduardo Villa por guiarme con su experiencia para culminar exitosamente el trabajo de titulación.

A la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO por formarme como profesional, y a la Fundación Amiga por darme la apertura para realizar este trabajo.

Erika

TABLA DE CONTENIDO

DEDICATORIA	iv
AGRADECIMIENTO	v
ÍNDICE DE ABREVIATURAS	viii
ÍNDICE DE TABLAS	x
ÍNDICE DE GRÁFICOS	xix
ÍNDICE DE ANEXOS	xx
RESUMEN	xxi
SUMMARY	xxii
INTRODUCCIÓN	1
CAPÍTULO I	7
1 MARCO TEÓRICO REFERENCIAL	7
1.1 Herramientas empleadas para el desarrollo del software	7
<i>1.1.1 PostgreSQL</i>	7
<i>1.1.2 Glassfish</i>	9
<i>1.1.3 StarUml</i>	10
<i>1.1.4 Power Designer</i>	11
<i>1.1.5 Framework</i>	12
<i>1.1.6 Patrón Modelo Vista Controlador</i>	12
<i>1.1.7 Java Server Faces</i>	13
<i>1.1.8 Prime Faces</i>	13
<i>1.1.9 Netbeans</i>	14
<i>1.1.10 Jasper Reports</i>	14
<i>1.1.11 IReport</i>	15
1.2 Calidad de software	15
<i>1.2.1 Calidad interna y externa del software</i>	15
<i>1.2.2 Norma ISO/IEC 9126</i>	16
1.3 Sistemas biométricos	19
<i>1.3.1 Ventajas de un sistema biométrico</i>	19
<i>1.3.2 Característica de un sistema biométrico para identificación personal</i>	20
<i>1.3.3 Sistemas biométricos basados en huellas dactilares</i>	20
1.4 Estado del Arte	22

CAPÍTULO II	25
2 MARCO METODOLÓGICO	25
2.1 Tipo de investigación	25
2.2 Métodos y técnicas	25
2.2.1 Metodología SCRUM	25
2.2.2 Técnicas	28
2.3 Herramientas	28
2.3.1 Herramientas para el desarrollo del software	28
2.3.2 Herramientas para la determinación de la eficiencia	29
2.3.3 Población	38
2.3.4 Muestra	38
CAPÍTULO III	39
3 RESULTADOS Y DISCUSIÓN	39
3.1 Desarrollo de un Sistema de control biométrico para los empleados de la “Fundación Amiga” de la ciudad de Esmeraldas, aplicando la norma ISO/IEC 9126-2 y 9126-3 para determinar la eficiencia del software.	39
3.1.1 Estudio Preliminar	39
3.1.2 Fase de Planificación	45
3.1.3 Fase de Desarrollo de Sprints	52
3.1.4 Fase de Cierre	89
3.2 Determinación de atributos no funcionales	91
3.2.1 Requisitos no funcionales	91
3.3 Determinación de la Eficiencia Interna y externa del sistema SISTAHU	92
CONCLUSIONES	131
RECOMENDACIONES	132
BIBLIOGRAFÍA	
ANEXOS	

ÍNDICE DE ABREVIATURAS

JSF	Java Server Faces
SW	Software
HW	Hardware
BD	Base de datos
IO	Entrada/Salida
SISTAHU	Sistema de talento humano
ODBC	Open DataBase Connectivity
JDBC	Java Database Connectivity
PHP	Preprocessor
TCL	Lenguaje de herramientas de comando
ANSI	Instituto Nacional Estadounidense de Estándares
SQL	Lenguaje de consulta estructurada
SHA	Algoritmo de Hash Seguro
XML	Lenguaje de Marcado Extensible
ACID	Atomicidad, Consistencia, Aislamiento y Durabilidad
GNU	GNU no es Unix
CDDL	Licencia Común de Desarrollo y Distribución
J2EE	Plataforma de Java, Edición Empresarial
MVC	Modelo Vista Controlador
HTTP	Protocolo de transferencia de hipertexto
HTML	Lenguaje de marcas de hipertexto
IDE	Entorno de desarrollo integrado

PDF	Formato de documento portátil
ISO	Organización Internacional de Normalización
IEC	Comisión Electrotécnica Internacional
AFIS	Sistema automático de identificación de huellas dactilares

ÍNDICE DE TABLAS

Tabla 1-2: Descripción de métrica interna Utilización I/O densidad de mensaje.....	32
Tabla 2-2: Descripción de métrica interna Utilización de memoria densidad de mensaje.....	33
Tabla 3-2: Descripción de métrica interna tiempo de respuesta.....	34
Tabla 4-2: Descripción de métrica externa tiempo de respuesta.....	35
Tabla 5-2: Descripción de métrica externa uso de memoria.....	36
Tabla 6-2: Descripción de métrica externa uso de procesador.....	37
Tabla 7-3: Prioridad de Ejecución.....	46
Tabla 8-3: Aplicación método T-Shirt.....	47
Tabla 9-3: Product Backlog.....	47
Tabla 10-3: Personas y roles.....	49
Tabla 11-3: Tipos y roles de usuarios.....	49
Tabla 12-3: Sprint Backlog.....	50
Tabla 13-3: Estudio de la Tecnología a utilizar.....	53
Tabla 14-3: Prueba de aceptación 1 para HT-01.....	53
Tabla 15-3: Tabla de Actividades para HT-02.....	54
Tabla 16-3: Tarea de Ingeniería 1 para la HT-02.....	54
Tabla 17-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	55
Tabla 18-3: Identificación del Riesgo.....	56
Tabla 19-3: Probabilidad del Riesgo.....	56
Tabla 20-3: Probabilidad del Riesgo.....	57
Tabla 21-3: Valoración del Impacto.....	57
Tabla 22-3: Valoración de la exposición del riesgo.....	58
Tabla 23-3: Exposición del riesgo.....	58
Tabla 24-3: Resumen riesgos.....	58
Tabla 25-3: HT-03 Diseño de la base de datos.....	65
Tabla 26-3: Prueba de aceptación 1 para HT-03.....	65
Tabla 27-3: Tabla de Actividades para HT-03.....	66
Tabla 28-3: Tarea de Ingeniería 1 para la HT-05.....	66
Tabla 29-3: Prueba de aceptación 1 para tara de ingeniería 1.....	67
Tabla 30-3: Tarea de Ingeniería 2 para la HT-03.....	68
Tabla 31-3: Prueba de aceptación 1 para tara de ingeniería 2.....	68
Tabla 32-3: Tarea de Ingeniería 3 para la HT-05.....	69
Tabla 33-3: Prueba de aceptación 1 para tara de ingeniería 3.....	69
Tabla 34-3: Tarea de Ingeniería 4 para la HT-05.....	70
Tabla 35-3: Prueba de aceptación 1 para tara de ingeniería 4.....	70
Tabla 36-3: HT-04 Diseño de la arquitectura del sistema.....	72
Tabla 37-3: Prueba de aceptación 1 para HT-04.....	72
Tabla 38-3: Tabla de Actividades para HT-04.....	73
Tabla 39-3: Tarea de Ingeniería 1 para la HT-04.....	73
Tabla 40-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	73
Tabla 41-3: HT-07 Estándar de codificación.....	77
Tabla 42-3: Prueba de aceptación 1 para HT-05.....	77
Tabla 43-3: Tabla de Actividades para HT-05.....	78
Tabla 44-3: Tarea de Ingeniería 1 para la HT-05.....	78

Tabla 45-3: Prueba de aceptación 1 para tarea de ingeniería 1	78
Tabla 46-3: HT-06 Estándar de codificación.	81
Tabla 47-3: Prueba de aceptación 1 para HT-06	81
Tabla 48-3: Tabla de Actividades para HT-06	81
Tabla 49-3: Tarea de Ingeniería 1 para la HT-06	82
Tabla 50-3: Prueba de aceptación 1 para tarea de ingeniería 1	82
Tabla 51-3: HT-07 Bocetos de la interfaz de usuario	84
Tabla 52-3: Prueba de aceptación 1 de la HT-07.....	85
Tabla 53-3: Tabla de Actividades para HU-07.....	85
Tabla 54-3: Tarea de Ingeniería 1 para la HT-09	86
Tabla 55-3: Prueba de aceptación 1 para tara de ingeniería 1	86
Tabla 56-3: Tarea de Ingeniería 2 para la HT-07	87
Tabla 57-3: Prueba de aceptación 1 para tarea de ingeniería 2.....	87
Tabla 58-3: Resultados Parámetros de Usabilidad	92
Tabla 59-3: Resultados Métrica Interna Utilización I/O Densidad de mensaje.....	93
Tabla 60-3: Resultados métrica interna utilización de memoria densidad de mensaje	99
Tabla 61-3: Resultados métrica interna tiempo de respuesta	107
Tabla 62-3: Resultados Métrica Externa Tiempo de Respuesta	108
Tabla 63-3: Resultados Métrica Externa Uso de Memoria Ram	115
Tabla 64-3: Resultados Métrica Externa Uso de Procesador	119
Tabla 65-3: Ponderación de las métricas de eficiencia interna y externa del SISTAHU	122
Tabla 66-3: Criterios de valoración para determinación de eficiencia de SISTAHU	123
Tabla 67-3: Valores obtenidos de utilización I/O densidad de mensaje	124
Tabla 68-3: Valores obtenidos de utilización de memoria densidad de mensaje	124
Tabla 69-3: Valores obtenidos de Tiempo de respuesta interno	125
Tabla 70-3: Valores obtenidos de uso de memoria ram.....	127
Tabla 71-3: Valores obtenidos de uso de procesador.....	128
Tabla 72-3: Valores obtenidos de tiempo de respuesta externo	128
Tabla 73-3: Resumen de las métricas de eficiencia interna y externa	129
Tabla 74: Diccionario de datos de Tabla asistencia	141
Tabla 75: Diccionario de datos de Tabla atraso.....	142
Tabla 76: Diccionario de datos de Tabla biometria	143
Tabla 77: Diccionario de datos de Tabla departamento	143
Tabla 78: Diccionario de datos de Tabla dia_registro	144
Tabla 79: Diccionario de datos de Tabla dirección_empleado.....	145
Tabla 80: Diccionario de datos de Tabla discapacidad_empleado	146
Tabla 81: Diccionario de datos de Tabla dispositivo	147
Tabla 82: Diccionario de datos de Tabla dispositivo_ubicación	148
Tabla 83: Diccionario de datos de Tabla empleado	148
Tabla 84: Diccionario de datos de Tabla empleado_horario_laborable.....	149
Tabla 85: Diccionario de datos de Tabla historial laboral.....	149
Tabla 86: Diccionario de datos de Tabla historial_periodo_empleado.....	150
Tabla 87: Diccionario de datos de Tabla horario_laborable.....	150
Tabla 88: Diccionario de datos de Tabla periodo	151
Tabla 89: Diccionario de datos de Tabla permiso	151
Tabla 90: Diccionario de datos de Tabla registro	152

Tabla 91: Diccionario de datos de Tabla solicitud_vacación	152
Tabla 92-3: Gestión de Riesgo 1	154
Tabla 93-3: Gestión de Riesgo 2	155
Tabla 94-3: Gestión de Riesgo 3	156
Tabla 95-3: Gestión de Riesgo 4	157
Tabla 96-3: Gestión de Riesgo 5	158
Tabla 97-3: HU-01 Ingreso de empleados	159
Tabla 98-3: Tabla de Actividades para HU-01	159
Tabla 99-3: Tarea de Ingeniería 1 para la HU-01	160
Tabla 100-3: Prueba de aceptación 1 para tarea de ingeniería 1	160
Tabla 101-3: Prueba de aceptación 2 para tarea de ingeniería 1	161
Tabla 102-3: HU-02 Actualización de Empleados	162
Tabla 103-3: Tabla de Actividades para HU-02	162
Tabla 104-3: Tarea de Ingeniería 1 para la HU-02	163
Tabla 105-3: Prueba de aceptación 1 para tarea de ingeniería 1	163
Tabla 106-3: Prueba de aceptación 2 para tarea de ingeniería 1	164
Tabla 107-3: HU-03 Lista de empleados	165
Tabla 108-3: Tabla de Actividades para HU-03	165
Tabla 109-3: Tarea de Ingeniería 1 para la HU-03	165
Tabla 110-3: Prueba de aceptación 1 para tarea de ingeniería 1	166
Tabla 111-3: Prueba de aceptación 2 para tarea de ingeniería 1	167
Tabla 112-3: Eliminación de Empleados	167
Tabla 113-3: Tarea de Ingeniería 1 para la HU-04	168
Tabla 114-3: Prueba de aceptación 1 para tarea de ingeniería 4	168
Tabla 115-3: Prueba de aceptación 2 para tarea de ingeniería 1	169
Tabla 116-3: HU-05 Reingreso de empleados	170
Tabla 117-3: Tabla de Actividades para HU-05	170
Tabla 118-3: Tarea de Ingeniería 1 para la HU-05	171
Tabla 119-3: Prueba de aceptación 1 para tarea de ingeniería 1	171
Tabla 120-3: Prueba de aceptación 2 para tarea de ingeniería 1	172
Tabla 121-3: HU-06 Actualización de Empleados	173
Tabla 122-3: Tabla de Actividades para HU-06	173
Tabla 123-3: Tarea de Ingeniería 1 para la HU-06	174
Tabla 124-3: Prueba de aceptación 1 para tarea de ingeniería 1	174
Tabla 125-3: Prueba de aceptación 1 para tarea de ingeniería 1	175
Tabla 126-3: HU-07 Lista de cumpleaños del mes	176
Tabla 127-3: Tabla de Actividades para HU-07	176
Tabla 128-3: Tarea de Ingeniería 1 para la HU-07	177
Tabla 129-3: Prueba de aceptación 1 para tarea de ingeniería 1	177
Tabla 130-3: Prueba de aceptación 2 para tarea de ingeniería 1	178
Tabla 131-3: HU-08 notificación de felicitación de cumpleaños	178
Tabla 132-3: Tabla de Actividades para HU-08	179
Tabla 133-3: Tarea de Ingeniería 1 para la HU-08	179
Tabla 134-3: Prueba de aceptación 1 para tarea de ingeniería 1	180
Tabla 135-3: Prueba de aceptación 2 para tarea de ingeniería 1	180
Tabla 136-3: HU-09 Ingreso de datos biométricos	181

Tabla 137-3: Tabla de Actividades para HU-09	181
Tabla 138-3: Tarea de Ingeniería 1 para la HU-09	182
Tabla 139-3: Prueba de aceptación 1 para tarea de ingeniería 1	182
Tabla 140-3: Prueba de aceptación 2 para tarea de ingeniería 1	183
Tabla 141-3: HU-10 Actualización de datos biométricos	184
Tabla 142-3: Tabla de Actividades para HU-10	184
Tabla 143-3: Tarea de Ingeniería 1 para la HU-10	185
Tabla 144-3: Prueba de aceptación 1 para tarea de ingeniería 1	185
Tabla 145-3: Prueba de aceptación 2 para tarea de ingeniería 1	186
Tabla 146-3: HU-11 Lista biometrías de empleados	187
Tabla 147-3: Tabla de Actividades para HU-11	187
Tabla 148-3: Tarea de Ingeniería 1 para la HU-11	188
Tabla 149-3: Prueba de aceptación 1 para tarea de ingeniería 1	188
Tabla 150-3: Prueba de aceptación 2 para tarea de ingeniería 1	189
Tabla 151-3: HU-12 Historial Laboral de empleados	189
Tabla 152-3: Tabla de Actividades para HU-04	190
Tabla 153-3: Tabla de Actividades para HU-04	190
Tabla 154-3: Prueba de aceptación 1 para tarea de ingeniería 1	191
Tabla 155-3: Prueba de aceptación 2 para tarea de ingeniería 1	191
Tabla 156-3: HU-13 Antigüedad Laboral	192
Tabla 157-3: Tabla de Actividades para HU-13	192
Tabla 158-3: Tarea de Ingeniería 1 para la HU-13	193
Tabla 159-3: Prueba de aceptación 1 para tarea de ingeniería 1	193
Tabla 160-3: Prueba de aceptación 2 para tarea de ingeniería 1	194
Tabla 161-3: HU-14 Actualización Historial Laboral de empleados	195
Tabla 162-3: Tabla de Actividades para HU-14	195
Tabla 163-3: Tabla de Actividades para HU-14	196
Tabla 164-3: Prueba de aceptación 1 para tarea de ingeniería 1	196
Tabla 165-3: Prueba de aceptación 2 para tarea de ingeniería 1	197
Tabla 166-3: HU-15 Sincronizar los registros diarios del empleado	197
Tabla 167-3: Tabla de Actividades para HU-15	198
Tabla 168-3: Tarea de Ingeniería 1 para la HU-15	198
Tabla 169-3: Prueba de aceptación 1 para tarea de ingeniería 1	199
Tabla 170-3: Prueba de aceptación 2 para tarea de ingeniería 1	199
Tabla 171-3: HU-16 Lista de Registro diario de empleados	200
Tabla 172-3: Tabla de Actividades para HU-16	200
Tabla 173-3: Tarea de Ingeniería 1 para la HU-16	201
Tabla 174-3: Prueba de aceptación 1 para tarea de ingeniería 1	201
Tabla 175-3: Prueba de aceptación 2 para tarea de ingeniería 1	202
Tabla 176-3: HU-17 Actualización de registros diarios de Empleados	203
Tabla 177-3: Tabla de Actividades para HU-17	203
Tabla 178-3: Tarea de Ingeniería 1 para la HU-17	204
Tabla 179-3: Prueba de aceptación 1 para tarea de ingeniería 1	204
Tabla 180-3: HU-18 Lista de asistencia de empleados	205
Tabla 181-3: Tabla de Actividades para HU-18	205
Tabla 182-3: Tarea de Ingeniería 1 para la HU-18	206

Tabla 183-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	206
Tabla 184-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	207
Tabla 185-3: HU-19 Lista de atrasos de empleados	208
Tabla 186-3: Tabla de Actividades para HU-19.....	208
Tabla 187-3: Tarea de Ingeniería 1 para la HU-19	209
Tabla 188-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	209
Tabla 189-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	210
Tabla 190-3: HU-20 Historial Laboral-Periodo de empleados.....	211
Tabla 191-3: Tabla de Actividades para HU-20.....	211
Tabla 192-3: Tarea de Ingeniería 1 para la HU-20	212
Tabla 193-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	212
Tabla 194-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	213
Tabla 195-3: HU-21 Registro de Solicitud de Vacaciones	214
Tabla 196-3: Tabla de Actividades para HU-21	214
Tabla 197-3: Tarea de Ingeniería 1 para la HU-21	215
Tabla 198-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	215
Tabla 199-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	216
Tabla 200-3: HU-22 Actualización de solicitud de vacaciones.....	217
Tabla 201-3: Tabla de Actividades para HU-22.....	217
Tabla 202-3: Tarea de Ingeniería 1 para la HU-22	218
Tabla 203-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	218
Tabla 204-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	219
Tabla 205-3: HU-23 Anulación de solicitud de vacación de empleados	220
Tabla 206-3: Tabla de Actividades para HU-23	220
Tabla 207-3: Tarea de Ingeniería 1 para la HU-23	220
Tabla 208-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	221
Tabla 209-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	222
Tabla 210-3: HU-24 Lista de solicitud de vacaciones activas de empleados	223
Tabla 211-3: Tabla de Actividades para HU-24.....	223
Tabla 212-3: Tarea de Ingeniería 1 para la HU-24	223
Tabla 213-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	224
Tabla 214-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	225
Tabla 215-3: HU-25 Generación de días calendario laborables de empleados	226
Tabla 216-3: Tabla de Actividades para HU-25	226
Tabla 217-3: Tarea de Ingeniería 1 para la HU-25	227
Tabla 218-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	227
Tabla 219-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	228
Tabla 220-3: HU-26 Actualización de días calendario laborables de empleados.....	229
Tabla 221-3: Tabla de Actividades para HU-26.....	229
Tabla 222-3: Tarea de Ingeniería 1 para la HU-26.....	230
Tabla 223-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	230
Tabla 224-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	231
Tabla 225-3: HU-27 Lista de días calendario laborables de empleados	232
Tabla 226-3: Tabla de Actividades para HU-27	232
Tabla 227-3: Tarea de Ingeniería 1 para la HU-27	232
Tabla 228-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	233

Tabla 229-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	234
Tabla 230-3: HU-28 Lista de días calendario laborables de empleados	234
Tabla 231-3: Tabla de Actividades para HU-28.....	235
Tabla 232-3: Tarea de Ingeniería 1 para la HU-28.....	235
Tabla 233-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	235
Tabla 234-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	236
Tabla 235-3: HU-29 Lista de días calendario laborables de empleados	237
Tabla 236-3: Tabla de Actividades para HU-29.....	238
Tabla 237-3: Tarea de Ingeniería 1 para la HU-29.....	238
Tabla 238-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	238
Tabla 239-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	239
Tabla 240-3: HU-30 Lista de días calendario laborables de empleados	240
Tabla 241-3: Tabla de Actividades para HU-30.....	241
Tabla 242-3: Tarea de Ingeniería 1 para la HU-30.....	241
Tabla 243-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	241
Tabla 244-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	242
Tabla 245-3: HU-31 Registro de horarios laborables de empleados	243
Tabla 246-3: Tabla de Actividades para HU-31.....	243
Tabla 247-3: Tarea de Ingeniería 1 para la HU-31.....	244
Tabla 248-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	244
Tabla 249-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	245
Tabla 250-3: HU-32 Actualización de horarios laborables de empleados	246
Tabla 251-3: Tabla de Actividades para HU-32.....	246
Tabla 252-3: Tarea de Ingeniería 1 para la HU-32.....	247
Tabla 253-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	247
Tabla 254-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	248
Tabla 255-3: HU-33 Lista de horarios laborables de empleados.....	249
Tabla 256-3: Tabla de Actividades para HU-33.....	249
Tabla 257-3: Tarea de Ingeniería 1 para la HU-33.....	250
Tabla 258-3: Prueba de aceptación 1 para tarea de ingeniería 1.....	250
Tabla 259-3: Prueba de aceptación 2 para tarea de ingeniería 1.....	251
Tabla 260-3: HU-34 Reporte de empleados clasificados por departamento	251
Tabla 261-3: Actividades de HU-34.....	252
Tabla 262-3: Tarea de Ingeniería para la HU-34.....	252
Tabla 263-3: Prueba de aceptación 1 para tarea de de ingenieria 1.....	253
Tabla 264-3: HU-35 Reporte de empleados no afiliados al IESS.....	254
Tabla 265-3: Actividades de HU-35.....	254
Tabla 266-3: Tarea de Ingeniería para la HU-35.....	255
Tabla 267-3: Prueba de aceptación 1 para tarea de de ingenieria 1.....	255
Tabla 268-3: HU-36 Reporte de asistencias por departamento	256
Tabla 269-3: Actividades de HU-36.....	256
Tabla 270-3: Tarea de Ingeniería para la HU-36.....	257
Tabla 271-3: Prueba de aceptación 1 para tarea de de ingenieria 1.....	257
Tabla 272-3: HU-37 Reporte de vacaciones gozadas, por gozar y saldos	258
Tabla 273-3: Actividades de HU-37.....	258
Tabla 274-3: Tarea de Ingeniería para la HU-37.....	259

Tabla 275-3: Prueba de aceptación 1 para tarea de de ingenieria 1	259
Tabla 276-3: HU-38 Reporte de vacaciones acumuladas	260
Tabla 277-3: Actividades de HU-38	260
Tabla 278-3: Tarea de Ingeniería para la HU-38	261
Tabla 279-3: Prueba de aceptación 1 para tarea de de ingenieria 1	261

ÍNDICE DE FIGURAS

Figura 1-1: Arquitectura Modelo Vista Controlador	13
Figura 2-1: Arquitectura Actual de ISO/IEC 14598 y 9126.....	17
Figura 3-1: Proceso de validación 1:1	21
Figura 4-1: Proceso de reconocimiento 1: N.....	21
Figura 5-2: Ejemplo de historia de usuario	26
Figura 6-2: Subcaracterísticas y métricas empleadas para la Eficiencia Interna	30
Figura 7-2: Subcaracterísticas y métricas empleadas para la Eficiencia Interna	31
Figura 8-3: Reloj biométrico de la Fundación Amiga	39
Figura 9-3: Controlador del reloj biométrico de la Fundación Amiga.....	40
Figura 10-3: Código para generar diccionario de datos	64
Figura 11-3: Diagrama de despliegue	71
Figura 12-3: Diagrama de Componentes	72
Figura 13-3: Diagrama de Clases “SISTAHU”	80
Figura 14-3: Bocetos de interfaz de usuario (Página Principal)	83
Figura 15-3: Pantalla Funcional General	84
Figura 16-3: Contador de Código SISTAHU	88
Figura 17-3: Controlador SISTAHU	94
Figura 18-3: Errores controlados	94
Figura 19-3: Error de empleado menor de edad	95
Figura 20-3: Error de duplicidad empleado	95
Figura 21-3: Error de inserción de empleado	96
Figura 22-3: Error al no contar con datos para guardar	96
Figura 23-3: Error al no contar con registros para vacaciones	97
Figura 24-3: Error al no contar con registros para atrasos	97
Figura 25-3: Error al no existe periodos activos.....	97
Figura 26-3: Error al no contar con registros para asistencia	98
Figura 27-3: Error al no existir periodos activos	98
Figura 28-3: Error al no contar con los estados del atraso	98
Figura 29-3: Errores con capturados mediante la sentencia TRY-CATCH.....	100
Figura 30-3: Error al crear o registrar el empleado, null.....	100
Figura 31-3: Error al crear dirección de empleado	101
Figura 32-3: Error al crear discapacidad del empleado	101
Figura 33-3: Error al crear cuenta empleado.....	101
Figura 34-3: Error al grabar dispositivos ubicación empleado.....	101
Figura 35-3: Error al intentar insertar registros externos a locales	102
Figura 36-3: Error al generar control biométrico vacaciones.....	102
Figura 37-3: Error al cargar días de vacaciones generadas	102
Figura 38-3: Error al intentar insertar registros externos a locales	103
Figura 39-3: Error al generar control biométrico vacaciones.....	103
Figura 40-3: Error al intentar cargar lista de atrasos generados	103
Figura 41-3: Error al intentar insertar registros externos a locales, asistencia	104
Figura 42-3: Error al intentar insertar registros externos a locales, vacaciones	104
Figura 43-3: Error al intentar cargar lista de asistencias generadas.....	104
Figura 44-3: Error al cargar el administrador de permisos	105

Figura 45-3: Error al guardar el permiso por tipo.....	105
Figura 46-3: Error al calcular número de días de permiso	105
Figura 47-3: Error al obtener datos del empleado	106
Figura 48-3: Tiempo de registro datos de un empleado.....	109
Figura 49-3: Tiempo de registro diario.....	110
Figura 50-3: Tiempo de traslado de información y generación de vacaciones, asistencia y atrasos.	110
Figura 51-3: Tiempo de visualización de registros diarios.	111
Figura 52-3: Tiempo de visualización días vacaciones generadas.	111
Figura 53-3: Tiempo de visualización de atrasos generados.....	112
Figura 54-3: Tiempo de visualización de asistencias generados.	113
Figura 55-3: Tiempo de registrar un permiso por tipo.....	113
Figura 56-3: Memoria usada registro datos del empleado	116
Figura 57-3: Memoria usada en generación los días de vacaciones a partir del registro diario	117
Figura 58-3: Memoria usada en atrasos a partir del registro diario	117
Figura 59-3: Memoria usada en asistencia a partir del registro diario	118
Figura 60-3: Memoria usada en registro permisos por tipo	118
Figura 61-3: Procesador usado en registro datos del empleado	120
Figura 62-3: Procesador usado en generación de vacaciones a partir del registro diario.....	120
Figura 63-3: Procesador usado en generación de atrasos a partir del registro diario.....	121
Figura 64-3: Procesador usado en generación de asistencia a partir del registro diario	121
Figura 65-3: Procesador usado en registro permisos por tipo	122

ÍNDICE DE GRÁFICOS

Gráfico 1-3: Proceso de contratación de un nuevo empleado	41
Gráfico 2-3: Proceso de Vacaciones	42
Gráfico 3-3: Proceso de Permisos	43
Gráfico 4-3: Proceso de turnos de trabajo	43
Gráfico 5-3: Proceso de traslado de un departamento a otro	44
Gráfico 6-3: Diagrama Entidad Relación	60
Gráfico 7-3: Diagrama lógico de la base de datos	62
Gráfico 8-3: Contador de Código "SISTAHU"	90
Gráfico 9-3: Gráfico de la ponderación para la eficiencia del SISTAHU	123
Gráfico 10-3: Gráfico de las métricas de la eficiencia interna del SISTAHU	126
Gráfico 11-3: Gráfico de las métricas de la eficiencia externa del SISTAHU	129
Gráfico 12-3: Gráfico de la eficiencia total obtenida mediante la norma ISO/IEC 9126-2 y 9126-3	130

ÍNDICE DE ANEXOS

ANEXO A: Plan de actividades	140
ANEXO B: Diccionario de Datos	141
ANEXO C: Gestión de Riesgos	154
ANEXO D: Historias de Usuario	159

RESUMEN

El objetivo del trabajo de titulación fue desarrollar un sistema de control biométrico para los empleados de la “Fundación Amiga” de la ciudad de Esmeraldas, aplicando la norma ISO/IEC 9126-2 y 9126-3 para determinar la eficiencia del software, el cual se lo realizó mediante la metodología ágil Scrum, utilizando las tres fases establecidas, en la fase de *planificación* se definieron los requerimientos funcionales y no funcionales, en la fase de *desarrollo de Sprints* se realizaron cada uno de los requerimientos definidos; diseños y funcionalidades del sistema, finalmente en la *fase de cierre* se realizó la evaluación de la gestión del proyecto, el cual está representado a través de un BurnDown Chart para evidenciar el nivel de cumplimiento de los sprints, además en esta gestión se determinó la productividad del proyecto en la misma que se obtuvo 227.78 lc/h. Para la determinación de la eficiencia interna y externa del software se aplicó la norma ISO/IEC 9126-2 Y 9126-3 de las cuales se tomó las subcaracterísticas *utilización de recursos* y *comportamiento temporal*, para la eficiencia interna las métricas de utilización de recursos empleadas fueron: Utilización I/O Densidad del mensaje, Utilización de memoria Densidad del mensaje, del mismo modo para el comportamiento temporal se utilizó el tiempo de respuesta, en cuanto a la eficiencia externa se analizó las métricas uso de memoria y uso del procesador para la subcaracterística utilización de recursos, de igual manera para el comportamiento temporal se utilizó la métrica tiempo de respuesta. Con el desarrollo del sistema la eficiencia del mismo cuenta con un 95%, basado en los parámetros establecidos en la norma especificada, de igual manera se pudo mejorar los procesos diarios de los empleados manteniendo la integridad de los datos; se recomienda investigar más a fondo sobre otros sistemas biométricos que se podrían implementar para el Sistema de Talento Humano (SISTAHU).

PALABRAS CLAVE: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <INGENIERÍA DE SOFTWARE>, <FUNDACIÓN AMIGA>, <CONTROL BIOMÉTRICO>, <NORMA ISO/IEC 9126-2>, <EFICIENCIA INTERNA>, <EFICIENCIA EXTERNA>, <NORMA ISO/IEC 9126-3>.

SUMMARY

The main goal of this research was to develop a biometric control system for the employees of “Fundación Amiga”, Esmeraldas City, by applying ISO/IEC standard 9126-2, and 9126-3 to determinate software’s efficiency, we used the agile methodology, SCRUM by using the three established standards, in the planning phase the functional and non-functional requirements were defined, in the development phase of sprints each of the defined requirements requested; designs and functionalities of the system, finally, in the closing phase, the evaluation phase of the Project management is carried out, which is represented by a BurnDown Chart to demonstrate the level of compliance of the sprints, in addition to this management, the productivity of the Project in the same that was obtained 227,78 lc/h. To determine the internal and external efficiency of the software we applied the standard ISO/IEC 9126-2 y 9126-3 of which the subcharacteristics, use of resources and temporal behavior were taaken, for the Message Density, in the same way for the temporal behavior we used the response time, as for the external efficiency we analyzed the metrics memory usage and processor use for the subcharacteristic resource utilization, like wise for the temporal behavior we used the metric time of response. With the development of the system, the efficiency of the system has 95% based on the parameters established in the specified standart, in the same way it was possible to improve the daily processes of the employees maintaining data integrity; Further research on other biometric systems that could be implemented for the Human Talent System (SISTAHU) is recommended.

KEYWORDS: Technology and sciencies of engineering, software engineering, Fundación Amiga, biometric control, ISO/IEC standard 9126-2, internal efficiency, external efficiency, ISO/IEC 9126-3.

INTRODUCCIÓN

Planteamiento del Problema

Antecedentes

Con el desarrollo de la tecnología se han ido incrementando herramientas informáticas que facilitan los procesos diarios de las personas. Gracias a ello las diferentes empresas que realizan funciones para brindar sus servicios se ven en la necesidad de adquirir productos software que se adapten a sus requerimientos y proporcionen rapidez y exactitud al realizar procesos.

El presente trabajo aplicativo se centraliza en el desarrollo de una aplicación web que permite el control de asistencia de los empleados de la Fundación Amiga, además se realizó un análisis de la eficiencia interna y externa de la aplicación, utilizando tecnología biométrica de huellas dactilares.

Fundación Amiga es una institución sin fines de lucro creada mediante acuerdo ministerial No.2634 del Ministerio de Educación y Cultura, el mismo que se encuentra publicado en el registro oficial No.971 del 3 de julio de 1992; cuyo fin es la ayuda social y la dignificación de las personas de bajos recursos económicos a través de artes y oficios, de la ciudad de Esmeraldas, sin distinción de edad, raza, cultura y religión.

Actualmente posee varias áreas que proporcionan resultados individualmente, como son fábrica de fideos, centro de salud privado, talleres de metal, mecánica y carpintería, escuela de Fútbol y Unidad Educativa San Daniel Conbonni.

La inserción de la Escuela de Fútbol y de la Unidad Educativa ha provocado el aumento a 80 personas que forman parte del personal administrativo, docente, de producción, etc. y con ello sale a la luz el exceso de trabajo y las deficiencias existentes en el manejo del personal. Esto ocasiona demora en los trámites de vacaciones, permisos, ingreso de información al IESS, lo que conlleva al incumplimiento de envío de información al Ministerio del trabajo.

Las entidades de control del trabajo han realizado multas por incumplimiento a la normativa, a la cual está obligada la institución en un estado de derecho. Además, se presentan quejas contra el departamento por parte de los administrados por errores en cuanto a la generación de sus documentos requeridos, así como exceso de trabajo solicitado a recursos humanos en un corto tiempo.

Esta institución que nació con un fin eminentemente social ha superado inconvenientes de toda índole, ahora la meta es superar los problemas existentes en el control de asistencia, permisos, vacaciones, etc. de los empleados, convirtiéndolo en eficiente y eficaz en el manejo.

Por ello se ha decidido solucionar esta deficiencia ya que no es posible continuar administrando la información de los trabajadores sin una herramienta sistemática que facilite el control, permisos y el cumplimiento de las jornadas de trabajo. Por lo cual se hace fundamental contar con una herramienta que ajustado a las necesidades de la institución facilite el control de la nómina y sea una base para la toma de decisiones.

Formulación del Problema

¿El Desarrollo de un sistema de control biométrico para los empleados de la “Fundación Amiga” de la ciudad de Esmeraldas, utilizando la norma ISO/IEC 9126-2 y 9126-3 permitirá mejorar la eficiencia en los procesos diarios del mismo?

Sistematización del Problema

- ¿Cómo se realiza actualmente el proceso de control de los empleados de la “Fundación Amiga”?
- ¿Cuáles son los requerimientos necesarios para el desarrollo del sistema de control biométrico?
- ¿Cuáles son las subcaracterísticas para determinar la eficiencia del software de acuerdo a la norma ISO/IEC 9126-2 y 9126-3?
- ¿Qué nivel de eficiencia alcanzará el proceso de control de empleados de la “Fundación Amiga” con el sistema propuesto?

Justificación

Justificación Teórica

Para el desarrollo de la aplicación se presentan las justificaciones necesarias exponiendo todas las razones que motivan a la realización del trabajo de titulación.

Debido a que la fundación no cuenta con una herramienta informática que agilice los procesos de control de la nómina de empleados se realizó este proyecto con la finalidad de apoyar en cada uno de ellos facilitando el manejo y revisión de los mismos para que de esta manera no ocurran errores

en cuanto a la generación y emisión de las acciones que incluyen los datos de la nómina de empleados.

Esta aplicación se realizó cumpliendo con los estándares de eficiencia expuestos en la norma ISO/IEC 9126-2 y 9126-3. De esta manera la calidad de este software satisface con la petición de los interesados de la misma.

La calidad de un producto software es el indicador que permite determinar si los procesos de construcción de software son apropiados. Es por esto que debe realizar una investigación e indagación sobre los métodos y técnicas que aseguren calidad en los productos, con el objetivo principal de generar propuestas determinadas para aplicaciones con características específicas. (Jiménez et al., 2014; p.9)

El sistema de control biométrico para los empleados de la “Fundación Amiga” de la ciudad de Esmeraldas, utilizando la norma ISO/IEC 9126-2 y 9126-3, está orientado a la optimización de la eficiencia interna y externa del software; es decir el tiempo de comportamiento temporal del sistema y utilización de recursos, para satisfacer los procesos de control del personal, por lo que se utilizó en esta aplicación las siguientes tecnologías.

Como gestor de base de datos PostgreSQL 9.5, el JDK 1.8 bajo el lenguaje de programación JAVA, IDE de desarrollo Netbeans versión 8.1, Framework JSF, con GlassFish 4.1.1 como servidor Web, a su vez se utilizó primefaces como herramienta del framework, para la realización de los reportes se empleó las herramientas Ireport versión 5.5 y JasperReports versión 4.5, de igual forma se aplicará SRCUM como metodología de desarrollo, y para determinar la eficiencia del software la norma ISO/IEC 9126-2 y 9126-3.

Se seleccionaron las herramientas tomando en cuenta en cuenta que: Scrum es una metodología ágil que permite obtener mejoras significativas, y en algunos casos un giro total en la productividad y la moral. Brinda un gran beneficio para los desarrolladores del producto software, ya que muchos de ellos están saturados por los constantes cambios de estilo en la gestión de proyectos. Scrum es simple y poderoso. (VASSdigital, 2012)

NetBeans IDE es un entorno de desarrollo integrado modular, de base estándar escrito en el lenguaje de programación Java, consiste en un IDE de código abierto y una plataforma de aplicación, las cuales pueden ser usadas como una estructura de soporte general (frameworks) para compilar cualquier tipo de aplicación. (Gimeno y Gonzales, 2011:pp.2-3)

PostgreSQL es un gestor de bases de datos orientado a objetos, muy conocido y usado en entornos de software libre porque cumple los estándares SQL92 y SQL99, y también por el conjunto de funcionalidades avanzadas que soporta. (Kasián y Reyes, 2012: p.4)

La principal razón por la que optamos por Netbeans como IDE de desarrollo y por PostgreSQL como SGBD (Sistema Gestor de Base de Datos) es que en la “Fundación Amiga” utilizan estas herramientas en los diferentes proyectos informáticos y fue solicitado aplicar estas herramientas en el desarrollo del sistema, además Netbeans junto a sus herramientas y pluggins ayudaron en el análisis estático de código, con lo cual se logró obtener una mejor eficiencia en desarrollo y calidad del código fuente.

Por otra parte, PostgreSQL en cuanto a calidad como SGBD (Sistema Gestor de Base de Datos) permitirá estabilidad y confiabilidad en el manejo de la información de nuestro sistema.

JSF Java Server Faces permite simplificar el desarrollo de aplicaciones Web, es un Framework que permite manejar el estado de los componentes, procesar los datos, validar la entrada del usuario, y manejar eventos. (Serrano y Minta, 2011: p.36)

Se utilizó el framework JSF porque como visión de nuestro proyecto es que el mismo sea escalable y el código JSF con el que creamos las vistas es muy parecido al HTML estándar. Lo pueden utilizar fácilmente desarrolladores y diseñadores web, además JSF es extensible, por lo que se pueden desarrollar nuevos componentes a medida.

Para complementar la utilización del framework JSF se empleó la herramienta Primefaces ya que esta cuenta con un gran número de componentes de OpenSource lo que facilita la creación de interfaces amigables además que cuenta con ThemeRoller Framework CSS que permite escoger entre 36 temas ya pre establecidos o generar nuevos con la herramienta online. (Flores, 2016, p.5)

A su vez que cumple a cabalidad dando solución a los requerimientos especificados, además de contar con una excelente documentación lo que facilita su manejo. (Flores, 2016, p.5)

Para la creación de los reportes se empleó JasperReports ya que nos facilita la generación de los mismos en diferentes formatos como son PDF, XML, HTML, CSV, XLS, RTF y TXT a través de pocas líneas de código, lo que permite almacenar su desarrollo en un archivo y posteriormente presentarlo a través de OutputStream, complementando a esto, existe un editor WYSIWYG el mismo que permite generar la parte visual de los reportes, llamada iReport. (Ambrocio, 2014, p.1)

Esta herramienta fue seleccionada debido a que se tiene conocimientos sólidos en el manejo de la misma facilitando su uso, como fue requerido por el presidente de la Fundación, además que esta herramienta nos permite editar formatos, tipos de letra, estilos, imágenes, decoraciones, etc.

El Patrón de diseño MVC es una guía para el desarrollo de arquitecturas de aplicaciones que ofrezcan una fuerte interactividad con los usuarios. Este patrón organiza la aplicación en tres modelos separados, **el primero** es un modelo que representa los datos de la aplicación y sus reglas de negocio, **el segundo** es un conjunto de vistas que representa los formularios de entrada y salida de información, **el tercero** es un conjunto de controladores que procesa las peticiones de los usuarios y controla el flujo de ejecución del sistema. (Carrera, 2014, pp.39-42)

Utilizamos la arquitectura MVC porque se ajusta a los requerimientos de la empresa, además se cuenta con mayor experiencia en el uso de esta arquitectura y de acuerdo a las exigencias de la “Fundación Amiga” se estableció utilizar las herramientas que más se adapten a nuestros conocimientos.

Para la determinación de la eficiencia del Software se aplicó la norma ISO/IEC 9126-2 y 9126-3, debido a que esta norma nos provee de métricas para determinar la eficiencia del software de una manera muy sencilla estableciendo criterios sólidos de evaluación del software, además de que esta norma es la base de otros estándares de calidad ya que de ella se derivan otros más actuales. (Ruiz et al., 2006: pp.25-29)

Para dar solución al problema que presenta la fundación el sistema cuenta con 3 módulos principales que son: Módulo de Gestión de Información del Trabajador, Módulo de Control y el Módulo de Reportes.

Justificación Práctica

Para efectuar la solución del problema planteado se creó una base de datos que permite almacenar toda la información de los empleados de la fundación, la cual podrá estar disponible para los interesados del negocio además de mantener la información disponible durante el ciclo de vida del software.

El Módulo de Gestión de Información del Trabajador consta de: gestión de datos personales, reingreso de datos del trabajador, traslado de personal de un departamento a otro, determinación del cumpleaños de cada trabajador enviando una notificación de felicitación a correo electrónico.

El Módulo de Control permite el registro de información biométrica del trabajador, registro de la hora ingreso y salida del trabajador, generación de atrasos del trabajador, generación de días de vacaciones de los trabajadores, gestión de solicitudes de vacaciones de los trabajadores, gestión de calendario de días no laborables y gestión de permisos por departamento siendo validado por talento humano y gestión de grupos de horarios laborables.

En cuanto a los reportes que se requirieron tenemos la nómina de trabajador clasificada por departamento, reporte de quienes no están afiliados al IESS, reporte de asistencia por departamento, reportes de vacaciones gozadas, por gozar y saldos y reporte de vacaciones acumuladas.

Este sistema garantiza la verificación de la asistencia de los empleados activos durante su jornada laboral, ya que mediante la huella dactilar se evita la impostura de identidad, así como permite facilitar la planificación y cumplimiento del cronograma de trabajo establecido en la “Fundación Amiga”

Objetivos

Objetivo General

Desarrollar un sistema de control biométrico para los empleados de la “Fundación Amiga” de la ciudad de Esmeraldas, aplicando la norma ISO/IEC 9126-2 y 9126-3 para determinar la eficiencia del software.

Objetivos específicos

- Describir los procesos actuales del control de los empleados de la “Fundación Amiga” de la ciudad de Esmeraldas.
- Definir los requerimientos del sistema de control biométrico.
- Diseñar e Implementar el sistema de control biométrico para la “Fundación Amiga” de la ciudad de Esmeraldas utilizando la norma ISO/IEC 9126-2 y 9126-3.
- Determinar la eficiencia interna y externa del sistema de control biométrico.

CAPÍTULO I

1 MARCO TEÓRICO REFERENCIAL

1.1 Herramientas empleadas para el desarrollo del software

Para llevar a cabo el desarrollo del sistema se utilizó ciertas herramientas de desarrollo de software, cada una pertenece al ámbito de software libre, debido a que fueron requeridas así por la Fundación Amiga, las mismas que fueron eficientes a la hora de llevar a cabo y cumplir con las funcionalidades esperadas del proyecto.

1.1.1 *PostgreSQL*

Existen un sin número de gestores de base de datos relacionales que permiten generar base de datos de alta eficiencia uno de ellos es Postgresql el mismo que es un motor de bases de datos relacionales que posee un gran reconocimiento por su fiabilidad, integridad de datos y correcto desempeño, además de que gracias a su máquina virtual integrada es portable lo que permite que no de problemas en los diferentes sistemas operativos como son Linux, Unix y Windows. (Kasián, y Reyes, 2012: p.4)

Se puede manejar sin ningún problema de paga ya que está bajo licencia de código abierto lo que permite ser usado, modificado y distribuido. (Kasián, y Reyes, 2012: p.4)

PostgreSQL proporciona varias características:

- Soporte para consultas con UNION, UNION ALL y EXCEPT.
- Outer Joins.
- Sub-selects.
- Cumple con ANSI SQL (implementando el estándar SQL92 y SQL99).

- Integridad referencial.
- Replicación (soluciones comerciales y no comerciales) que permiten la duplicación de bases de datos maestras en múltiples sitios de réplica.
- Interfaces nativas para ODBC, JDBC, C, C++, PHP, Perl, TCL, ECPG, Python y Ruby.
- Reglas.
- Vistas.
- Procedimientos almacenados.
- Triggers.
- Unicode.
- Secuencias.
- Herencia: Incluye herencia entre tablas por ello se lo clasifica como gestor objeto relacional.
- Una API abierta.
- Soporte nativo SSL.
- Lenguajes procedurales.
- Respaldo en caliente.
- Bloqueo a nivel.
- Índices parciales y funcionales.
- Autenticación Kerberos nativa.
- Extensiones para SHA1, MD5, XML y otras funcionalidades.
- Herramientas para generar SQL portable para compartir con otros sistemas compatibles con SQL.
- Sistema de tipos de datos extensible para proveer tipos de datos definidos por el usuario, y rápido desarrollo de nuevos tipos.
- Funciones de compatibilidad para ayudar en la transición desde otros sistemas menos compatibles con SQL.

- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP, etc.), cadenas de bits entre otros. También permite la creación de tipos propios.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
- Cumple completamente con ACID (ACID es un acrónimo de Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad en español).
- Replicación Sincrónica: permitiendo alta disponibilidad con consistencia sobre múltiples servidores.
- Regionalización por columna: soportando correctamente el ordenamiento por lenguaje en las bases de datos, tablas o columnas.
- Tablas unlogged: importante incremento del rendimiento para datos efímeros.
- Nivel de Aislamiento Serializable a través de "Snapshots": mantiene consistentes múltiples transacciones concurrentes sin el uso de bloqueos, usando verdadera serialización.
- Writeable Common Table Expressions: ejecuta actualizaciones multi-fases complejas en una simple consulta.
- Security-Enhanced Postgres: despliega seguridad de nivel militar y control de acceso mandatorio.

Indexamiento de los k vecinos más cercanos (k-Nearest-Neighbor): índices basados en distancias para consultas rápidas de ubicación y búsquedas de texto. (Kasián, y Reyes, 2012: pp.4-5)

1.1.2 Glassfish

1.1.2.1 Qué es Glassfish

Este término fue acuñado por sus creadores basados en un pez que es transparente es decir se puede ver sus huesos, esto fue tomado ya que querían que el proyecto fuera transparente además de que utiliza una licencia Open Source, específicamente la licencia Common Development and Distribution License (CDDL) v1.0 y la GNU Public License (GPL) v2. (Serra, 2010, p.96)

1.1.2.2 Para qué sirve Glassfish

Es un servidor de aplicaciones, además de ser una comunidad de consumidores que permite desplegar aplicaciones que han sido especificadas en la plataforma Java EE, fue creado por Sun Microsystems. Cuenta con una versión comercial llamada Sun GlassFish Enterprise Server. Al ser una comunidad libre, sus interesados son capaces de incorporar nuevas características, además de realizar testeos para verificar las fallas y lograr una correcta funcionalidad. (Serra, 2010,96)

1.1.2.3 Cómo funciona un servidor de aplicaciones

Otorga funcionalidades built in, lo que evita escribir código fuente. Esto es gracias a que sus componentes se construyen dentro de un ambiente de ejecución virtual llamado dominio de ejecución. Verifica si el desarrollador tiene los permisos necesarios para invocar a un determinado método dentro de los beans y sus implementaciones. (Serra, 2010, p.97)

1.1.3 StarUml

1.1.3.1 Qué es y para qué sirve

Es una herramienta muy popular en el ámbito de desarrollo de software ya que permite modelar diagramas que ayudan en el proceso de ingeniería de software para documentar el flujo de trabajo de una determinada aplicación. (Canepa, 2012,pp.57-67)

1.1.3.2 Características

- Permite convertir un diagrama en otro de forma automatizada.
- Soporta la documentación de los objetos y diagramas, lo que facilita elaborar los informes y aspectos formales del proyecto.
- Cuenta con una unidad que le permite convertir los diagramas de secuencia en colaboración y viceversa. (Canepa, 2012, pp-57-67)

1.1.4 Power Designer

1.1.4.1 Qué es Power Designer

Es una herramienta software que permite modelar, planificar y gestionar los metadatos del software que constituyen la arquitectura de un determinado conjunto de datos, información y de la empresa. Esta herramienta permite realizar modificaciones en el diseño de arquitecturas complejas las veces que se consideren necesarias. (SAP Power Designer, 2017)

Brinda a los encargados de crear bases de datos y dirigentes de la misma al mantenimiento heterogéneo sólido para cada una, y brinda servicios de modelado de datos, tecnología de enlace y sincronización, gestión de metadatos y mucho más. (SAP Power Designer, 2017)

1.1.4.2 Capacidades funcionales

- Vincular el negocio y las tecnologías de información, mediante la tecnología de enlace y de sincronización se pueden implementar vínculos entre los requerimientos, sintaxis de los negocios, y patrones de datos. (SAP Power Designer, 2017)
- Optimizar la eficiencia de los desarrolladores de la empresa, mediante el almacenamiento de la información y comunicación de la misma con un almacén de metadatos en donde se encuentran varios tipos de modelado. (SAP Power Designer, 2017)
- Permite almacenar la información de cada uno de los sistemas ya creados, para tener a la mano dicha documentación del sistema. (SAP Power Designer, 2017)
- Consiente un ambiente open source, ya que mejora su comprensión de los sistemas heterogéneos con un mantenimiento de alto nivel para base de datos, lenguaje XML, etc. (SAP Power Designer, 2017)

1.1.5 Framework

Es un conjunto de componentes que pueden ser manipulables es decir una aplicación genérica la cual es un prototipo para generar una aplicación completa y eficiente. (Gutiérrez, 2014, p.1)

1.1.6 Patrón Modelo Vista Controlador

Para entender el funcionamiento de los frameworks es necesario conocer el Patrón MVC. El patrón Modelo-Vista-Controlador es un mapa para diseñar arquitecturas de sistemas que permitan una comunicación eficiente con usuarios. (Gutiérrez, 2014, p.2)

Este patrón consta de tres capas o modelos, el primero representa los diferentes datos del sistema, así como las reglas del negocio, el segundo corresponde a las interfaces que presenciara el usuario que comprenden los formularios de entrada y salida de información, el tercero comprende controladores que manipulan los requerimientos de los usuarios y verifica el flujo de ejecución del sistema. (Gutiérrez, 2014, p.2)

La Filosofía de diseño de las diferentes aplicaciones. reúne lo siguiente como se puede visualizar en la **figura 1-1**. Modelo: Corresponde la funcionalidad central, el eje y estado de la aplicación. Trabaja en su medio, no comprende ninguna dependencia con el controlador y la vista. (Flores, 2016, p.5)

Vista: Permite visualizar el Modelo y ser modificada cuando exista una variación en éste, ya que tiene estrecha relación con el mismo para acceder a él, sin embargo se restringe la modificación del estado del modelo. (Flores, 2016, p.5)

Controlador: Emite una respuesta a la petición del usuario, estableciendo la operación y instaurando el modelo. (Flores, 2016, p.6)

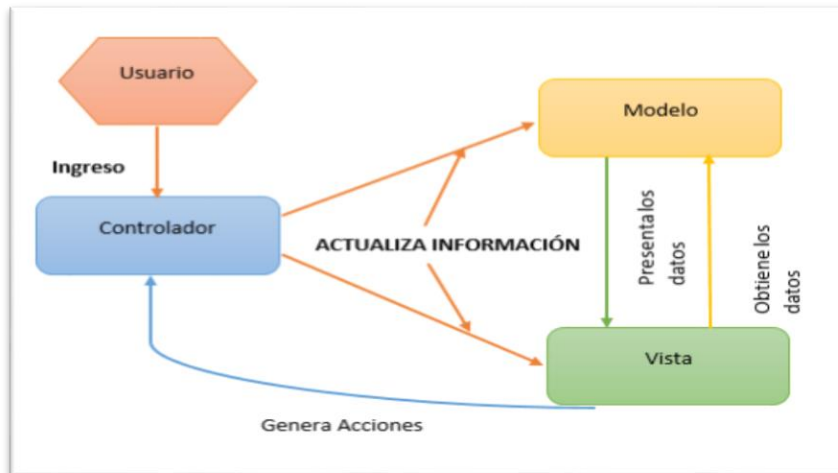


Figura 1-1: Arquitectura Modelo Vista Controlador
Realizado por: Erika R. 2017

1.1.7 *Java Server Faces*

Se conocen un sin número de frameworks orientados a diferentes ámbitos.

JSF es un framework web J2EE orientado a la interfaz de usuario proveniente de la familia de código libre, el mismo que se basa en componentes de interfaz de usuario que pueden ser incluidos en páginas dinámicas, eventos e interacciones entre ellos, que permiten desarrollar aplicaciones web de una manera más simple y rápida. Además, que no muestra al programador Datos de entrada, salida de HTTP ni markups HTML, obteniendo una visión de programación parecida a las aplicaciones de escritorio. (Díaz et al., 2007: p.3)

Brinda diversos componentes que simplifican la generación de vistas como son: botones, tablas, calendarios, etc. (Flores, 2016, p.4)

1.1.8 *Prime Faces*

Es una librería de componentes visuales de código abierto para el conjunto Java Server Faces 2.0 desarrollada y mantenida por Prime Technology.

Este brinda una serie de componentes que permiten facilitar el desarrollo y delineación de aplicaciones web. (Pech et al., 2010: p.8)

Las principales características de PrimeFaces son:

- Soporte nativo de Ajax, incluyendo Push/Coment.
- Kit para crear aplicaciones web móviles.
- Es compatible con otras librerías de componentes como Jboss RichFaces.
- Uso de JavaScript no intrusivo.
- Es un proyecto open source, activo y estable. (Pech et al., 2010: p.8)

1.1.9 Netbeans

Netbeans IDE es un entorno integrado de desarrollo modular con base estándar, desarrollado bajo plataforma Java, además el proyecto Netbeans consiste en un IDE de código abierto y una plataforma de aplicación, las cuales pueden ser usadas como una estructura de soporte general (framework) para compilar cualquier tipo de aplicación. (Oracle Corporation and/or its affiliates, 2017)

Permite el acoplamiento de los lenguajes de programación con las plataformas de los sistemas operativos, lo que facilita el diseño y progreso de una aplicación ya sea móvil, web o de escritorio, brindando a los desarrolladores eficacia en cuanto a la creación, reajuste, compilación, refinación, prueba e implantación de las aplicaciones creadas. (Oracle Corporation and/or its affiliates, 2017)

1.1.10 Jasper Reports

Es una biblioteca que permite la generación de informes referente a la información que se solicite. Ha sido creado en lenguaje Java, permitiendo emplear los diferentes datos que se obtengan de diversas fuentes y desarrollar documentos que puedan ser visualizados, impresos o exportados en un sin número de formatos como: HTML, PDF, Excel, Open Office y Word. JasperReports trabaja

frecuentemente en conjunto iReport, un front-end gráfico de código abierto el cual permite la edición de los reportes. (Flores, 2016, p.101)

1.1.11 IReport

Es un editor para JasperReport desarrollado netamente en Java. Asistiendo a los desarrolladores en la generación de reportes de forma visual, esto es debido a la interfaz con una amplia usabilidad que: otorga una conectividad efectiva a una base de datos y un editor de consultas Querys para la creación de reportes tanto generales como personalizados, además incluye la generación de gráficos con JfreeChart. (Henríquez, 2008,pp.24-26)

Esta es una librería que permite generar gráficos como son: pastel, barras, 3d, burbujas, etc. Los reportes generados podrán emplearse en diferentes plataformas, además tienen una exposición con alta formalidad, en varios formatos: PDF, XML, HTML, etc. (Henríquez, 2008,pp.24-26)

1.2 Calidad de software

ISO en la norma 8402:1994, la define como la “Totalidad de propiedades y características de un producto, proceso o servicio que le confiere su aptitud para satisfacer unas necesidades expresadas o implícitas.” (ISO-8402:1994,2016), en la actualización la Norma ISO, 9000:2015, la define como el “Grado en el que un conjunto de características inherentes cumple con los requisitos”. (Vera, 2011, p.30)

1.2.1 Calidad interna y externa del software

1.2.1.1 Calidad interna del software

Su objetivo es el mejoramiento de los procesos internos de una empresa. Mediante ésta se implementan medios que permiten la excelencia en la descripción posible de la organización y detectar y limitar los funcionamientos incorrectos. Los favorecidos de la calidad interna son tanto la administración como los empleados de la compañía. (ADELACEN, 2013, p.IV)

La calidad interna pasa generalmente por una etapa participativa en la que se identifican y formalizan los procesos internos. Tiene como objetivo medir la calidad del software mediante

factores medibles durante su desarrollo. Los factores propios que se encuentran dentro del proceso se enfocan en las particularidades constructivas de los componentes, debido a que solo se puede acceder a ellos y ser controlados por sus fabricantes. (Carvallo et al., 2010: p.11)

1.2.1.2 Calidad externa del software

Tiene como objetivo la satisfacción de los clientes mediante el cumplimiento total del producto software que satisfaga las expectativas del mismo. (ADELACEN LA, 2013, p.IV)

Los favorecidos de la calidad externa son tanto los clientes como los socios externos de una empresa. Ya que este tipo de operaciones necesita para cumplirse con éxito escuchar a los clientes y además permitirá que se tomen en cuenta cada una de las necesidades implícitas que los clientes no determinar literalmente. (ADELACEN LA, 2013, p.IV)

1.2.2 Norma ISO/IEC 9126

Es un estándar internacional que permite la evaluación de la calidad del software. Esta se divide en 4 partes:

ISO/IEC 9126-1: Este comprende el modelo de calidad, el mismo que especifica las características y subcaracterísticas que son empleadas por las partes siguientes del estándar.

ISO/IEC 9126-2: Este comprende las métricas externas.

ISO/IEC 9126-3: Este comprende las métricas internas.

ISO/IEC 9126-4: Este comprende la calidad en uso. (Ortega y Villavicencio, 2011:p.2)

La serie original de la norma ISO / IEC 9126 se dividió en 9126 y 14598 como se puede visualizar en la **figura 2-1**, ya que aquel modelo de calidad y las métricas son útiles no sólo para la evaluación del producto sino también para la especificación de requisitos de calidad. La evaluación de la calidad es posible y de vital importancia sólo cuando los requisitos de calidad están claramente especificados. (Azuma, 2001, pp.337-346)

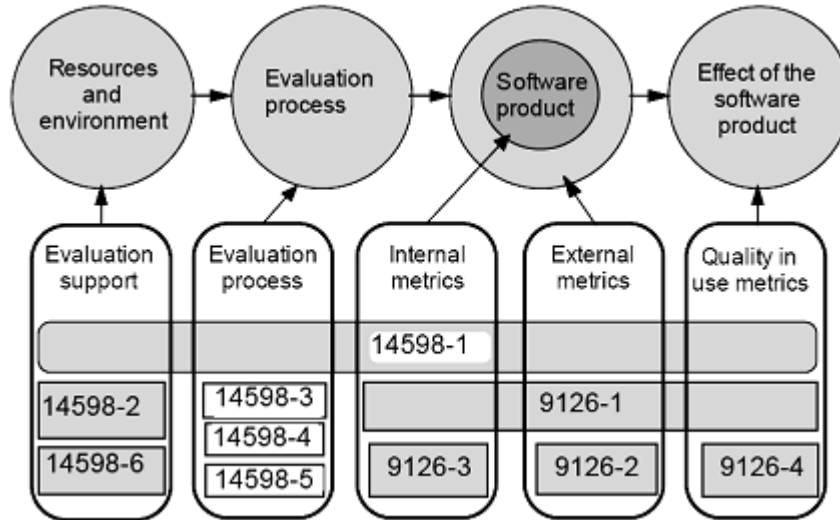


Figura 2-1: Arquitectura Actual de ISO/IEC 14598 y 9126.
Fuente: (Azuma, 2001, pp.337-346)

1.2.2.1 Métricas de eficiencia externa establecidas en la norma ISO/IEC 9126-2

Estas métricas deben medir de forma efectiva atributos como el gasto del tiempo y los recursos que se utilizan, así como el comportamiento del sistema, ya sea durante las pruebas o las diferentes opciones que se requiera. (Vivanco, 2011, p.20)

Para llevar a cabo esta métricas es necesario anotar la función que realizan los factores como el uso del procesador y la memoria utilizada por otro software, tráfico de red y procesos que se ejecutan en segundo plano. Las posibles incertidumbres y los parámetros definidos como permitidos para datos obtenidos deben ser determinados y contrastados con las diferentes especificaciones de requerimientos. (NTP-ISO/IEC-TR 9126-2, 2004, p.4)

Métricas de comportamiento temporal: Estas métricas deben medir el tiempo del comportamiento del sistema de cómputo, así como el producto mientras está en pruebas y además cuando ya está listo. (Vivanco, 2011, p.30)

Las métricas que propone el estándar son:

- **Tiempo de respuesta:** Mide el tiempo que tarda en culminar una tarea además del tiempo en recibir una respuesta de esa tarea. (Vivanco, 2011, p.30)

Métricas de utilización de recursos: Estas métricas deben medir la utilización de recursos, comportamiento del sistema de cómputo, así como el producto mientras está en pruebas y además cuando ya está listo. (Vivanco, 2011, p.31)

Las métricas que propone el estándar son:

- **Utilización de recurso de dispositivo de E/S:** Mide la utilización de dispositivos de entrada y salida. (Vivanco, 2011, p.31)
- **Utilización de recursos de memoria:** Mide la máxima utilización de la memoria del pc. (Vivanco, 2011, p.31)

1.2.2.2 Métricas de eficiencia interna establecidas en la norma ISO/IEC 9126-3

Estas métricas se usan para conjeturar la eficiencia del comportamiento del producto software mientras se realiza el testeo y su ejecución. Se debe definir las condiciones donde se realizará la medición de la eficiencia. Por ejemplo: cuando se compra un software este deberá tener sus especificaciones de implantación para un correcto funcionamiento. (NTP-ISO/IEC-TR 9126-3, 2005, p.8)

- **Métricas de comportamiento temporal:** Este tipo de métricas nos reflejan un conjunto de características y atributos para predecir el tiempo en cómo se comporta el sistema durante las pruebas u operaciones. (Vivanco, 2011, pp.40-41)

Las métricas que propone el estándar son:

- **Tiempo de respuesta:** Se refiere al tiempo esperado para culminar con una tarea determinada. (Mena,2006, p.13)
- **Métricas de utilización de recursos:** Este tipo de métricas nos reflejan un conjunto de características y atributos que permiten predecir la utilización de los recursos hardware donde se esté ejecutando el sistema. (Vivanco, 2011, pp.40-41)

Las métricas que propone el estándar son:

- **Utilización I/O densidad de densaje:** Mide la densidad de mensajes de acuerdo al uso de entradas y salidas en las líneas de código responsables de esos procesos.

- **Utilización de memoria Densidad de Mensaje:** Mide la densidad de mensajes de acuerdo al uso de memoria en las líneas de código responsables de esos procesos. (Vivanco, 2011, pp.123-124)

1.3 Sistemas biométricos

Los sistemas biométricos proporcionan un papel indispensable en los procesos de reconocimiento de personas, los mismos que están basados en políticas públicas de seguridad. En general los gobiernos se fundamentan en la biometría para identificar a las personas, autenticar su identidad en sistemas informáticos, reforzar la seguridad pública en aeropuertos y ciudades, y restringir el acceso a sitios seguros, tanto físicos como virtuales. (Etchart et al., 2011: p.2)

Los sistemas biométricos, en general, están compuestos de dispositivos para recopilar datos en formato digital; algoritmos de procesamiento de los datos recolectados, que efectúan control de calidad de los datos y van conformando las plantillas biométricas. (Etchart et al., 2011: p.1)

Las plantillas se almacenan en una base de datos contra la cual se comparan los datos ingresantes en la posterior actividad de verificación. El proceso de comparación permite realizar el cotejo de los nuevos datos con los datos almacenados. Finalmente, un proceso de decisión recomienda tomar una decisión en el proceso de reconocimiento, a partir de los resultados del componente de coincidencia. (Etchart et al., 2011: p.1)

1.3.1 Ventajas de un sistema biométrico

- Identificación inequívoca y exclusiva del personal.
- El código de identificación es personal. Lo que garantiza que solo la persona autorizada es reconocida como tal.
- Evita la pérdida y olvido del código biométrico, ya que cada persona autorizada cuenta en sí con dicho código.
- Un sistema de huellas dactilares genera cero costos de mantenimiento, así como en enseres. (Chacha et al., 2013: p.28)

1.3.2 Característica de un sistema biométrico para identificación personal

Debe cumplir con ciertas características que permitirán que el sistema sea seguro y confiable de usar estas son: desempeño, aceptabilidad y fiabilidad. (Chacha et al., 2013: p.25)

Desempeño: Se enfoca básicamente en la exactitud, la rapidez y la robustez lograda en la verificación del personal a través del sistema biométrico. El objetivo de esta característica es demostrar que el sistema desarrollado brinde una exactitud y rapidez admisible. (Chacha et al., 2013: p.25)

Aceptabilidad: Se refiere a la aceptación cotidiana que tendrá el Sistema Biométrico por parte de los usuarios, presentando facilidad de uso y comodidad al emplearlo, es decir; que le inspire confianza. (Chacha et al., 2013: p.25)

Fiabilidad: Representa la integridad que tendrá el sistema en cuanto al reconocimiento biométrico, ya que actualmente existen tecnologías que permiten crear dedos de látex, grabaciones digitales de voz, prótesis de ojos, entre otros, para eludir la seguridad del sistema y de esta manera acceder sin restricción al objetivo deseado. (Chacha et al., 2013: p.25)

1.3.3 Sistemas biométricos basados en huellas dactilares

La huella dactilar es un distintivo que tiene una mayor acogida en comparación con otras tecnologías biométricas que se presentan en la actualidad. Debido a esto su uso es adquirido con amplia fiabilidad ya que dos dedos nunca serán iguales; ya sean de gemelos o de una misma persona. y de esta manera determinar la identidad de los individuos en áreas donde se requiere una mayor seguridad como son: policial, civil y comercial. (Aching y Rojas, 2003: pp.11-20)

Un sistema automático de identificación dactilar o AFIS (Automated Finger print Identification System), es un sistema de reconocimiento de huellas dactilares, sin embargo, lo que hacen realmente es comparar y buscar similitudes. Los AFIS son capaces de trabajar con un amplio número de individuos, estos clasifican las huellas dactilares de acuerdo a las redes dactilares que contienen ya que constan de motores expertos en procesar información, de esta manera el desarrollo del reconocimiento dactilar se realiza en varios procesos facilitando el mismo. (Arrieta et al., p.1)

Para ello el sistema biométrico captura la imagen, el sistema mediante un proceso de normalización corrige la posición del mismo, extrae las características, la posición de cada una de las minucias y las compara con las que tiene en su base de datos para autorizar el acceso. En la base de datos se

almacenan varias imágenes y de esta forma efectúa la incluye como nueva imagen o consulta si ha sido validada (1:1) **figura 3-1** o reconocida (1: N) **figura 4-1**. (Reina y Tapia, 2012: p.14)

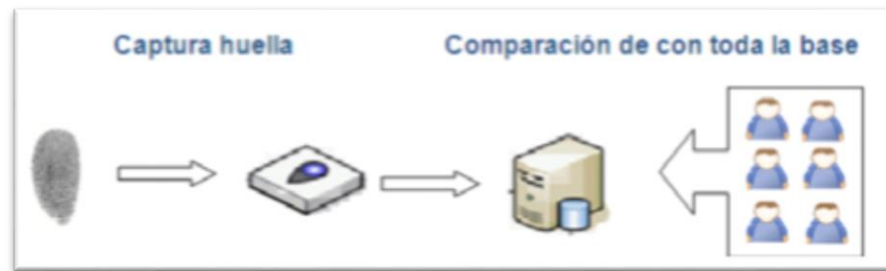


Figura 3-1: Proceso de validación 1:1
Fuente: (Reina y Tapia, 2012: p.14)



Figura 4-1: Proceso de reconocimiento 1: N
Fuente: (Reina y Tapia, 2012: p.14)

Para reconocer si dos huellas dactilares pertenecen o no a un individuo es necesario realizar un proceso que comienza con la categorización de la huella dactilar y finaliza con el balance de los detalles primordiales de ambas huellas. La categorización de huellas concierne a un estudio a escala “gruesa” de los diferentes esquemas generales de la huella que garantizan la asignación a una clase determinada, lo que implica en una parte o fraccionamiento de la base de datos a ser revisada. (Arrieta et al.,p.2)

Sin embargo, la concordancia de huellas acarrea un examen a escala “fina” de las huellas dactilares iniciando el proceso desde los vectores de particularidades resultantes, los cuales figuran la geometría de cada punto especificado. (Arrieta et al.,p.2)

1.4 Estado del Arte

Para el desarrollo de este proyecto se han tomado en cuenta varias investigaciones que se han llevado a cabo dentro del campo de la Ingeniería de Software entre las que se destacan:

“Medición de la Calidad de Productos de Software en un Ambiente Académico Usando la Norma ISO/IEC 9126”

Mediante el presente estudio, los alumnos de los cursos de ingeniería de software de la Escuela Superior Politécnica del Litoral presentaron el uso de la norma ISO/IEC 9126 para valorar la calidad de los productos de software desarrollados por ellos. Cada uno de estos productos se desarrolló para organizaciones ecuatorianas asentadas en la ciudad de Guayaquil. (Ortega y Villavicencio, 2011: p.2)

Este estudio determinó que la evaluación de la calidad debe seguir un modelo de referencia, ya sea una norma o a su vez un modelo que la organización construya, lo que permite facilitar el desarrollo de la evaluación del software que se requiere. Se realizó un registro de las diferentes mediciones obtenidas, estas fueron determinadas por los integrantes de cada grupo; las métricas fueron: tiempo de desarrollo, tiempo de corrección, tiempo total y de proceso. (Ortega y Villavicencio, 2011: p.4)

Se evaluó la calidad interna y externa del software a través de ciertas características como son la eficiencia, mantenibilidad, usabilidad, fiabilidad, funcionalidad además de otras características de calidad de uso como son la satisfacción y la eficiencia. Para determinar cada uno de estos parámetros se realizó una matriz en donde se colocaban los valores según el grado de importancia de cada una de ellas. (Ortega y Villavicencio, 2011: p.4)

Con ello se pudo concluir que la funcionalidad y eficiencia obtuvieron porcentajes mayoritarios. Estas características fueron determinadas de acuerdo a las peticiones de los clientes ya que contaban con una incertidumbre en cuanto al software que obtendrían de sus estudiantes. (Ortega y Villavicencio, 2011: p.4)

“Análisis de los modelos de calidad de software existentes y su apoyo al cumplimiento de requerimientos de empresas no dedicadas al desarrollo de software”

En este estudio se planteó determinar la capacidad de conocimiento que tienen las diferentes empresas que adquieren software en las metodologías para el aseguramiento de la calidad de un determinado software. Con los resultados de este se creó un modelo de calidad. (Espinosa et al., 2010: p.VI)

Para la mejora de calidad existen diversos ámbitos de actuación: Nivel Individual, Nivel de empresa u organización y el Nivel de proyecto

Además, se enfoca en determinar las dos partes en las que se debe detallar un sistema de calidad las cuales son Documentación y una Parte Práctica. (Espinosa et al., 2010: p.19)

En esta investigación se aplicó la encuesta como instrumento de medición debido a que es más rápido para determinar resultados. Con ella también se logra fácilmente realizar un análisis estadístico respectivo así como el debido tratamiento informático para obtener diversos resultados que se requieran. (Espinosa et al., 2010: p.68)

Con este estudio se logró determinar que las empresas en general se enfocan más en los costos o la experiencia con los desarrolladores de software y no ponen su respectivo énfasis en la calidad que tendrá el software requerido. (Espinosa et al., 2010: p.126)

“Evaluación de calidad del sistema integrado para casas de valores sicav de la bolsa de valores de Quito utilizando la norma ISO/IEC 14598”

El presente estudio se enfoca en determinar la Calidad de Software, además que se estudia el Estándar de Modelo de Calidad de Software ISO/IEC 9126 ya que este modelo es el que se centra al estándar de proceso de Evaluación ISO/IEC 14598.

Además, se establece detalladamente el estudio del modelo de calidad escogido ya que este se aplica para el caso propuesto. (Vivanco, 2011, pp.1-3)

Así mismo se analiza que seleccionar una buena política permite alcanzar la calidad del software que se espera, pero no la asegura, es por esto que se necesita un control en el ciclo de vida posterior a su producción. (Vivanco, 2011, p.3)

Para realizar el análisis de los resultados se aplicaron fórmulas que vienen dadas en la norma a usarse lo que permitió obtener que el sistema en estudio cumple con un 82% de calidad, debido a que se encuentra dentro de la parametrización es considerado como un PRODUCTO SATISFACTORIO, es decir que satisface los requisitos como base que se determinaron al sistema implementado. (Vivanco, 2011, p.175)

De esta forma se especifica que el sistema en estudio tiene una alta usabilidad gracias al conjunto de manuales, con una documentación entendible y precisa, lo que otorga al sistema la característica de ser fácil de usar. (Vivanco, 2011, p.181)

“Análisis, diseño y desarrollo de un sistema de administración control y monitoreo de equipos de autenticación biométrica.”

En esta investigación se contempla el desarrollo de un sistema automatizado, que optimice y facilite su proceso de control de asistencia y pago de nómina de sus empleados. Para ello se presenta el desarrollo de un “prototipo de sistema el mismo que se utiliza para manejar eficazmente la asistencia y a su vez pago de la nómina de los empleados”. (Reina y Tapia, 2012: p.78)

Dicho sistema utiliza los resultados de la tecnología biométrica para asegurarse de que la información sea completa y veraz, toda esa información es guardada en una base de datos en SQL, y utiliza los DTS para actualizar la información cada cierto tiempo. (Reina y Tapia, 2012: p.4)

Este sistema se ejecutará sobre diferentes lectores biométricos ya que el usuario no olvidará su código biométrico, es decir; su mano o su ojo, etc. que tienen una complejidad mayor a la hora de impostar su identidad, este tipo de dispositivos garantizan mayor seguridad debido a la deducción de que cada persona es única con rasgos físicos exclusivos. (Reina y Tapia, 2012: p.18)

CAPÍTULO II

2 MARCO METODOLÓGICO

2.1 Tipo de investigación

La presente investigación es una investigación de tipo aplicada, ya que permite aplicar metodologías de desarrollo de software, estándares, características y métricas existentes en la fundación Amiga, para generar nuevos procesos informáticos que mejoran los existentes.

2.2 Métodos y técnicas

2.2.1 Metodología SCRUM

2.2.1.1 Definición

Se aprecia como una metodología ágil y flexible, ya que define un proceso de desarrollo iterativo e incremental que puede ser aplicado a cualquier producto software o la gestión de diferentes actividades complejas, proporcionando una sólida integración entre los equipos de desarrollo. Lo que permite enlazarlo con la participación activa de los clientes, aumentando que los requisitos y las peticiones del cliente se entiendan más rápidamente. (Bissi, 2007, pp.3-6)

2.2.1.2 Elementos

Existen elementos esenciales que hacen posible una metodología Scrum estos son:

Producto Backlog: Comprende el conjunto de los requerimientos del cliente este puede ser organizado como una lista de necesidades. (Trigas, 2012, p.37)

Esta lista la desarrollará el cliente guiado por el Scrum Master, el mismo que estimará el costo de cada requisito para ser culminado con éxito, además de lo que añada un valor final al producto software. (Trigas, 2012, p.37)

- Contendrá los objetivos del Producto, en cada objetivo se indicará el valor que le da el cliente y el coste estimado; de esta manera se realiza la lista, priorizando por valor y coste.
- Se indicará las iteraciones y los entregables que propone el cliente.
- Además. Los riesgos y la gestión de los mismos. (Trigas, 2012, p.37)

Las historias de usuario:

Describen las funcionalidades que tendrá el software, las mismas que irán evolucionando a lo largo del proyecto.

Están formadas por 3 fases:

- **Card:** Es corta descripción
- **Conversation:** Es una conversación que servirá para aseverar el entendimiento general y sintetizar el objetivo.
- **Confirmation:** Pruebas funcionales para establecer puntos importantes. (Trigas, 2012, p.38)

En la **figura 5-1** se puede visualizar un ejemplo de historia de usuario con cada una de sus partes con su respectiva descripción. (Trigas, 2012, p.37)

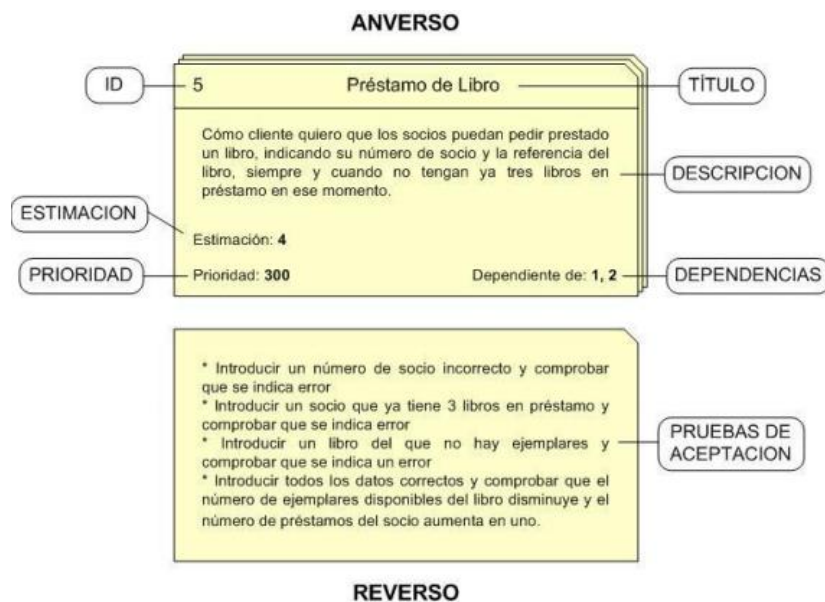


Figura 5-2: Ejemplo de historia de usuario
Fuente: (Trigas, 2012, p.38)

Formato de la Pila: La documentación constante es menos estricta ya que se plantea mantener una comunicación directa con los miembros del equipo, debido a esto maneja el Backlog como herramienta. (Trigas, 2012, p.39)

Se debe incluir:

- Identificar para la funcionalidad
 - Descripción de la funcionalidad
 - Sistema de priorización u orden
 - Estimación. (Trigas, 2012, p.39)
-
- **Sprint Backlog:** Comprende el conjunto de actividades que se llevarán a cabo en un Sprint. Permite asignar tareas a cada integrante del equipo para fragmentar el proyecto en secciones más pequeñas para determinar lo que está fallando y dar solución al mismo. (Trigas, 2012, p.40)

Se debe incluir:

- Lista de Tareas
 - Responsable de la tarea, el estado y tiempo por terminar de la misma.
 - Incorpora consulta diaria de los miembros del equipo.
 - Incorpora referencias para determinar el tiempo restante de cada tarea. (Trigas, 2012, p.40)
-
- **Incremento:** Es una sección finalizada y lista para funcionar del sprint. (Trigas, 2012, p.41)

2.2.1.3 *Ciclo de vida*

- **Fase de Planificación:** Definición de una nueva versión basada en la pila actual, junto con una estimación de coste y agenda. Si se trata de un nuevo sistema, esta fase abarca tanto la visión como el análisis. Si se trata de la mejora de un sistema existente comprende un análisis de alcance más limitado. Arquitectura: Diseño de la implementación de las funcionalidades de la pila. Esta fase incluye la modificación de la arquitectura y diseño generales. (Scrum Manager, 2013)

- **Fase de Desarrollo de sprints:** Desarrollo de la funcionalidad de la nueva versión con respeto continuo a las variables de tiempo, requisitos, costo y competencia. La interacción con estas variables define el final de esta fase. El sistema va evolucionando a través de múltiples iteraciones de desarrollo o sprints. (Scrum Manager, 2013)
- **Fase de Cierre:** Preparación para el lanzamiento de la versión, incluyendo la documentación final y pruebas antes del lanzamiento de la versión. Cuando el equipo de gestión siente que las variables de tiempo, parte completada, requisitos, coste y calidad están alineadas para producir una nueva versión, declaran cerrada la versión, dando paso a esta fase. (Scrum Manager, 2013)

En esta fase se prepara el producto generado para producir una nueva versión. Entre las tareas de cierre se encuentran: integración, pruebas del sistema, documentación de usuario, preparación del material de formación y marketing. (Scrum Manager, 2013)

2.2.2 Técnicas

La técnica empleada fue la observación; ya que a través de ella se obtuvo información relevante en cuanto a los procesos que se llevan a cabo para el control de la asistencia, así como la generación y emisión de órdenes de permisos y vacaciones a los empleados.

Otra de las técnicas empleadas fue la revisión de documentación en la cual se analizó la normativa interna de la “Fundación Amiga”, en donde están detallados los procesos de gestión de información de empleados.

De igual manera para la determinación de la eficiencia del software se aplicó diversas pruebas con softwares específicos los cuales se especificarán en el siguiente apartado.

2.3 Herramientas

2.3.1 Herramientas para el desarrollo del software

Power Designer 16.5: Diseño lógico de la base de datos

Visio 2016: Creación del modelo entidad relación

Star Uml 5.0.2: Creación del diagrama de clases

Bizagi Modeler 2.7.0.2: Para el modelado de procesos actuales de la Fundación Amiga

Netbeans 8.1: Desarrollo de SISTAHU

Java: 1.8.0: Lenguaje de Programación

Windows 10: Versión del Sistema Operativo

Postgresql 9.5: Gestor de Base de datos

Glass Fish 4.1.1: Servidor web

Java Server Faces 2.2: Framework de desarrollo

Primefaces 5.3: Biblioteca de componentes para el framework

2.3.2 *Herramientas para la determinación de la eficiencia*

Para la determinación de la eficiencia interna y externa se utilizó las métricas establecidas en la norma ISO/IEC 9126-2 y 9126-3 en la cual se especifica el siguiente formato de medición del recurso.

- Nombre de la métrica
- Propósito de la métrica
- Método de aplicación
- Medición, formula y cálculo de datos
- Interpretación de los valores medidos
- Tipo de escala de métrica
- Tipo de Medida
- Entradas para medición
- Usuarios Seleccionados

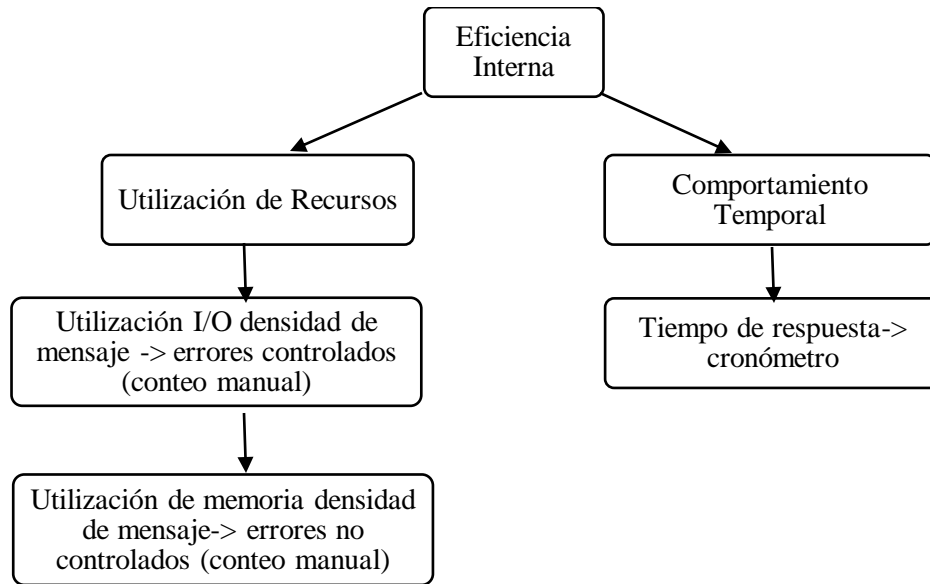


Figura 6-2: Subcaracterísticas y métricas empleadas para la Eficiencia Interna

Realizado por: Erika Reina. 2017

En la **Figura 6-2** se puede visualizar las dos subcaracterísticas analizadas en cuanto a la eficiencia interna, estas fueron: utilización de recursos y comportamiento temporal.

En la utilización de recursos se empleó la métrica la utilización I/O densidad de mensaje a través del conteo manual de los errores controlados dentro de los procesos a medirse y en la utilización de memoria densidad de mensaje a través del conteo manual de los errores no controlados, de igual manera en el comportamiento temporal se midió el tiempo de respuesta en cada proceso a través de un cronómetro.

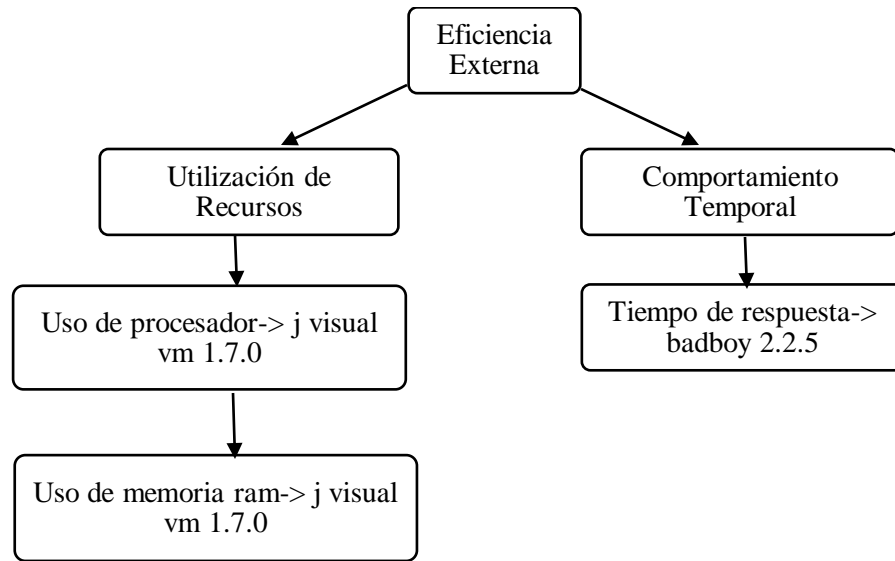


Figura 7-2: Subcaracterísticas y métricas empleadas para la Eficiencia Interna

Realizado por: Erika Reina. 2017

Así mismo en la **figura 7-2** se puede visualizar que para la eficiencia externa se utilizó el comportamiento temporal en el cual se midió el tiempo de respuesta a través de la aplicación **badboy 2.2.5** de igual manera se analizó la utilización de recursos en donde se midió el uso de memoria y uso del procesador mediante la aplicación **j visual vm 1.7.0**.

A continuación, se describen los formatos con su especificación de cada una de las métricas que se llevaron a cabo para la determinación de la eficiencia.

Eficiencia Interna

Característica: Eficiencia **Subcaracterística:** Utilización de recursos.

Tabla 1-2: Descripción de métrica interna Utilización I/O densidad de mensaje

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala de métrica	Tipo de Medida	Entradas para medición	Usuarios Seleccionados
Utilización I/O Densidad de mensaje	¿Cuál es la densidad de mensajes relacionado con la utilización de I/O en las líneas de código responsables hacienda llamadas del Sistema?	Contar el número de errores que pertenecen a fallas de I/O, advertencias y comparar al número estimado de líneas de código responsable en llamadas al sistema	$X=A/B$ A=Número de I/O relacionados con mensajes del error B=Número de líneas de código directamente relacionados con llamadas del sistema	El mayor El mejor	Absoluto	X=contable/contable A=contable B=contable	Código Fuente	Diseñadores

Fuente: (Vivanco Villamar, 2011: p.111)

Se utilizó evaluaciones de código de Entrada y Salida. Las evaluaciones se realizan mediante instrucciones IF y se las conoce como errores controlados. Y el conteo de líneas de código de cada proceso se realizó de forma manual.

Característica: Eficiencia **Subcaracterística:** Utilización de recursos.

Tabla 2-2: Descripción de métrica interna Utilización de memoria densidad de mensaje

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala de métrica	Tipo de Medida	Entradas para medición	Usuarios Seleccionados
Utilización de Memoria Densidad de mensaje	¿Cuál es la densidad de mensajes relacionado con la utilización de memoria en las líneas de código responsables haciendo llamadas del Sistema?	Contar el número de mensajes de errores que pertenecen al fallo de memoria y advertencia y comparar el número estimado de líneas de código responsable en llamadas del sistema	$X=A/B$ A=Número de memoria relacionada con los mensajes de error B=Número de líneas de código directamente relacionadas a las llamadas del sistema	El mayor El mejor	La proporción	X=contable/contable A=contable B=contable	Código Fuente	Diseñadores

Fuente: (Vivanco Villamar, 2011: p.112)

Se evaluó únicamente con una instrucción general denominada TRY-CATCH; son errores relacionados al lenguaje de programación y al uso de memoria. De igual manera el conteo de líneas de código de cada proceso se realizó de forma manual.

Característica: Eficiencia

Subcaracterística: Comportamiento Temporal

Tabla 3-2: Descripción de métrica interna tiempo de respuesta

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Tiempo de respuesta	¿Cuál es el tiempo estimado para completar una tarea?	Evaluar la eficiencia de las llamadas al SO y a la aplicación. Estimar el tiempo de respuesta basado en ello.	$X =$ tiempo (calculado o simulado)	Entre más corto, mejor	Proporción	$X =$ tiempo	Sistema Operativo conocido. Tiempo estimado en llamadas al sistema.	Usuarios desarrolladores

Fuente: (Mena, 2006: p.13)

Para la estimación de este tiempo se utilizó un cronómetro iniciado en 0 el cual fue detenido al culminar cada una de las tareas especificadas.

Eficiencia Externa

Característica: Eficiencia

Subcaracterística: Comportamiento Temporal

Tabla 4-2: Descripción de métrica externa tiempo de respuesta

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y de cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Tiempo de respuesta	¿Cuánto tiempo le ha tomado terminar una tarea específica? ¿Cuánto tiempo le toma recibir una respuesta a las tareas específicas?	Empiece una tarea especificada. Mida el tiempo que toma para la muestra para terminar su operación. Guarde un registro de cada intento.	T= Tiempo calculado	$0 < T$ El más temprano es el mejor	Ratio	X=tiempo	Reporte de prueba. Informe de la operación mostrada en un lapso de tiempo	Usuarios desarrolladores

Fuente: (Vivanco Villamar, 2011: p.123)

Para la ejecución de esta métrica se empleó la herramienta Badboy versión 2.2.5 que captura a través de la carga de la aplicación a su entorno, los tiempos requeridos al ejecutar las diferentes operaciones solicitadas.

Característica: Eficiencia

Subcaracterística: Utilización de Recursos

Tabla 5-2: Descripción de métrica externa uso de memoria

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y de cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Uso de Memoria	¿Cuánta memoria ha utilizado para terminar una tarea específica?	Empiece una tarea especificada. Mida la cantidad de memoria que toma para terminar su operación. Guarde un registro de cada intento.	$X =$ Memoria ocupada	$0 < X$ El menor es el mejor	Ratio	$X =$ Mega Bytes	Reporte de prueba. Informe de la operación mostrada en capacidad de memoria usada	Usuarios desarrolladores

Realizado por: Erika Reina, 2017

Fuente: (Vivanco Villamar, 2011: p.112)

Para la ejecución de esta métrica se empleó la herramienta J Visual Vm versión 1.7.0 que captura a través de la carga de la aplicación a su entorno, el uso de memoria ram, al ejecutarse una operación específica.

Característica: Eficiencia

Subcaracterística: Utilización de Recursos

Tabla 6-2: Descripción de métrica externa uso de procesador

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y de cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Uso de Procesador	¿Cuánto procesador ha utilizado para terminar una tarea específica?	Empiece una tarea especificada. Mida la cantidad de procesador que toma para terminar su operación. Guarde un registro de cada intento.	$X =$ Porcentaje promedio de uso	$0 < X$ El menor es el mejor	Ratio	$X =$ Porcentaje de uso	Reporte de prueba. Informe de la operación mostrada en porcentaje de uso	Usuarios desarrolladores

Realizado por: Erika Reina. 2017

Fuente: (Vivanco Villamar, 2011: p.111)

Para la ejecución de esta métrica se empleó la herramienta J Visual Vm versión 1.7.0 que captura a través de la carga de la aplicación a su entorno, el uso del procesador, al ejecutarse una operación específica.

2.3.3 Población

Para realizar las mediciones que determinaron la eficiencia del software se tomó una población de 38 requerimientos funcionales establecidos para el sistema SISTAHU, los mismos que se encuentran definidos como historias de usuario.

2.3.4 Muestra

Se tomó una muestra de la población de cinco requerimientos funcionales. Tomando en cuenta los requerimientos que tienen un mayor uso en el sistema como son:

- Ingresar datos del empleado
- Generar los días de vacaciones a partir del registro diario
- Generar atrasos mediante el registro diario del empleado
- Generar asistencia a partir del registro diario del empleado
- Registrar permisos por tipo

CAPÍTULO III

3 RESULTADOS Y DISCUSIÓN

3.1 Desarrollo de un Sistema de control biométrico para los empleados de la “Fundación Amiga” de la ciudad de Esmeraldas, aplicando la norma ISO/IEC 9126-2 y 9126-3 para determinar la eficiencia del software.

3.1.1 Estudio Preliminar

Fundación Amiga es una institución privada creada mediante acuerdo ministerial No.2634 del Ministerio de Educación y Cultura, cuyos procesos se encuentran regidos bajo el código de trabajo.

Fundación Amiga realiza sus procesos de control de empleados mediante un reloj biométrico de una antigüedad de 7 años, el mismo que al momento de realizar el análisis de la situación actual no contaba con información de sus características técnicas; esto fue debido a que la persona que lo instaló ya no se encontraba trabajando en la institución y no se maneja información al respecto.

En la **figura 8-3** se puede visualizar el reloj biométrico de la fundación en funcionamiento.



Figura 8-3: Reloj biométrico de la Fundación Amiga

Realizado por: Erika Reina. 2017

Fuente: Fundación Amiga. 2017

El reloj biométrico cuenta con un controlador que permite descargar la información almacenada en el mismo, en un ordenador ubicado en el departamento de recursos humanos. Este software suele tener problemas en la descarga de información, ya que la aplicación suele colgarse, ocasionando que el reloj biométrico deje de funcionar.

En la **figura 9-3** podemos visualizar las características del controlador del reloj biométrico

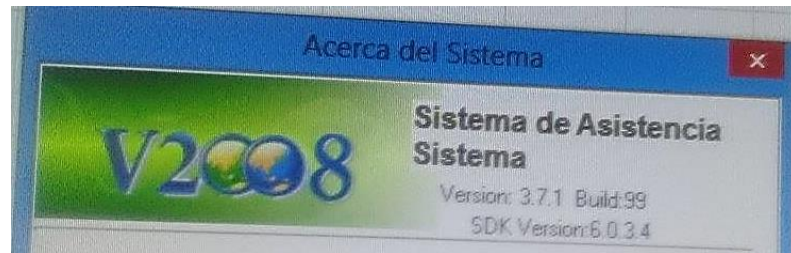


Figura 9-3: Controlador del reloj biométrico de la Fundación Amiga

Realizado por: Erika Reina. 2017

Fuente: Fundación Amiga. 2017

Desarrollo de los procesos de control de los empleados de la fundación Amiga

Departamento de talento humano

Contratación de un empleado

Después de realizar la contratación del nuevo empleado, se registran los datos personales, profesionales y laborales en un archivo de Excel, se lo ingresa al IESS, y una vez registrado el contrato en MRL se archiva la documentación en la carpeta de cada empleado por departamentos. (Batioja, 2017) Esto se lo puede visualizar en el **gráfico 1-3**.

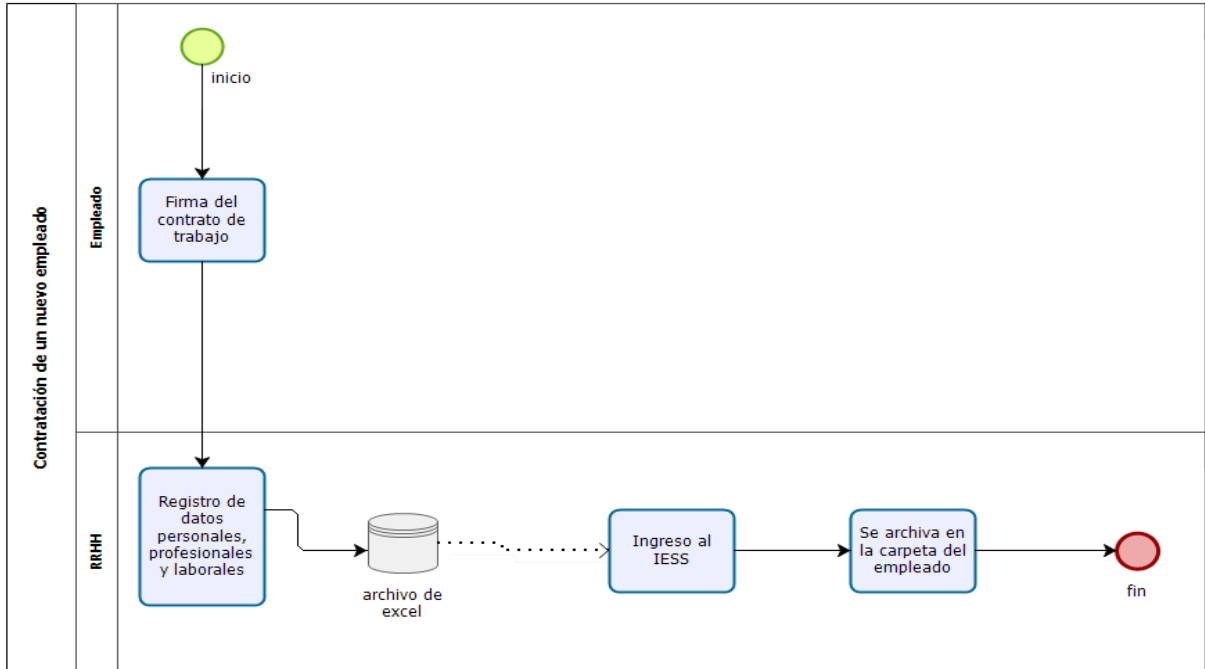


Gráfico 1-3: Proceso de contratación de un nuevo empleado
 Realizado por: Erika R. 2017

Vacaciones

Control de vacaciones

El ingreso del calendario de vacaciones y modificación del calendario, se lo realiza al iniciar el año en un archivo de Excel. (Batioja, 2017)

Goce de vacaciones

El control de los días disponibles y restantes que le quedan al empleado se lleva en Excel y se realiza la resta respectiva para determinar los días a gozar de las vacaciones. (Batioja, 2017)

En el **gráfico 2-3** se puede visualizar el proceso de vacaciones desde el ingreso del calendario hasta el almacenamiento y uso de la información guardada.

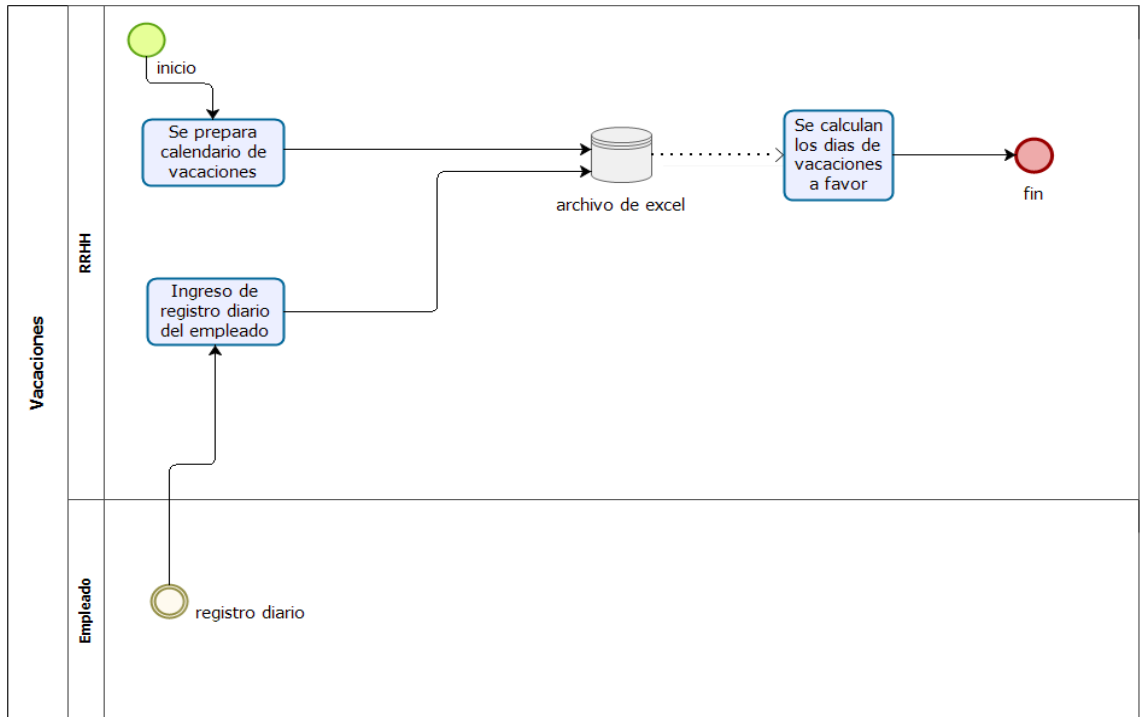


Gráfico 2-3: Proceso de Vacaciones
 Realizado por: Erika R. 2017

Permisos

Control de salidas, multas, etc.

En el **gráfico 3-3** se puede determinar que cada empleado se controla en una carpeta en donde cada vez que hay una afectación de multas, permisos, etc. se la coloca en la carpeta para su historial. (Batioja, 2017)

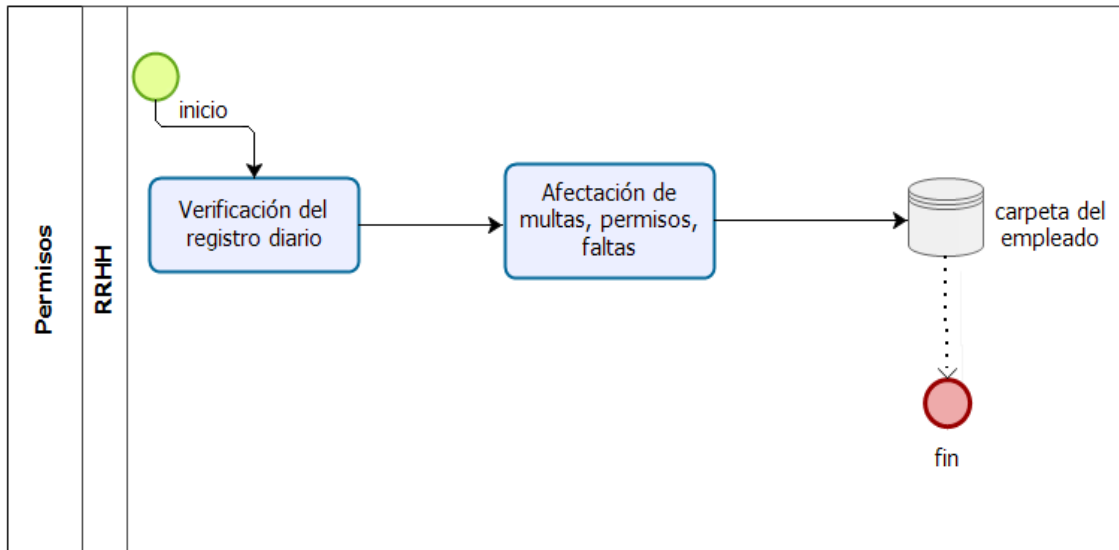


Gráfico 3-3: Proceso de Permisos

Realizado por: Erika R. 2017

Registro de turnos de trabajo

El gráfico 4-3 nos presenta que los cronogramas de turno de trabajo, cambio de turnos de trabajo se lo realiza en Excel, modificando la información, horarios, etc. (Batioja, 2017)

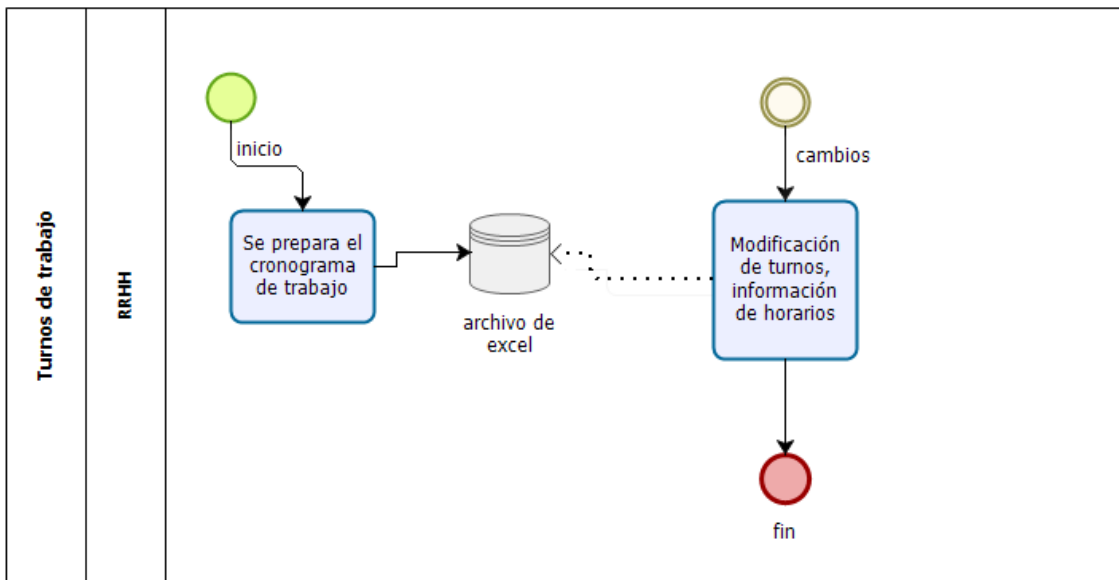


Gráfico 4-3: Proceso de turnos de trabajo

Realizado por: Erika R. 2017

Traslado de un departamento a otro

Cuando el empleado es trasladado a otro departamento se realiza mediante el archivo de Excel donde esta registrado su departamento de trabajo, colocando su información en otra sección del departamento destino. (Batioja, 2017) Esto se lo puede visualizar en el **gráfico 5-3**.

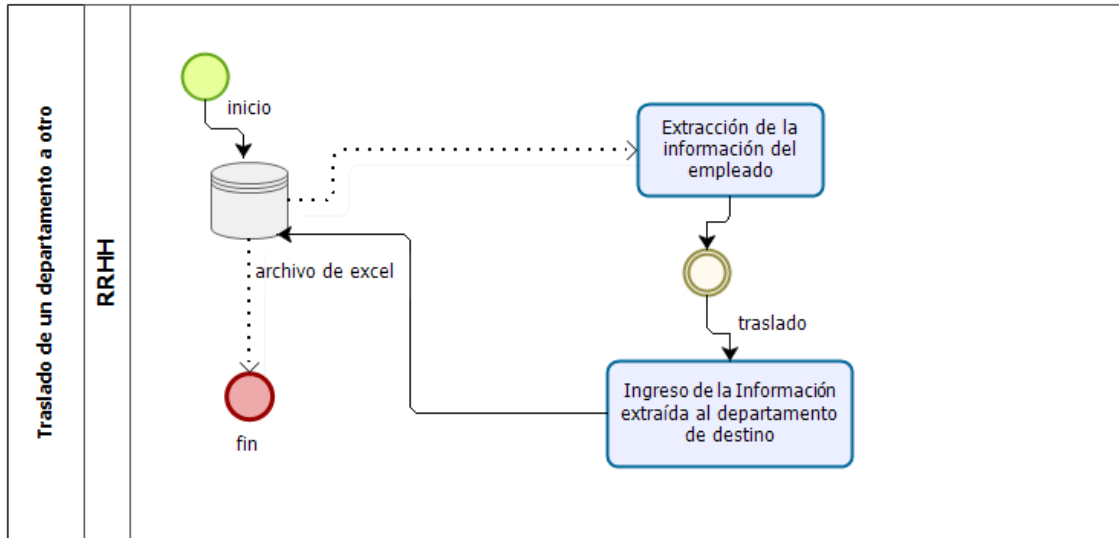


Gráfico 5-3: Proceso de traslado de un departamento a otro
Realizado por: Erika R. 2017

La eficiencia en el control de los empleados de la Fundación Amiga, tiene un gran déficit ya que muchos de los procesos se realizan de forma manual por lo que existe pérdida de la Información e inconsistencia de la misma, de igual forma se realizan procesos individuales en Excel el cual no tiene la misma rapidez y funcionalidad que un sistema de control biométrico que permite gestionar estos procesos de mejor manera. (Batioja, 2017)

3.1.2 Fase de Planificación

Con el fin de lograr una calendarización y una óptima coordinación de las tareas planteadas, se realiza la planificación donde se elaboran los sprints de acuerdo a la prioridad que tenga cada requerimiento, las metáforas del sistema no son solicitadas por el usuario, pero son complemento importante en el desarrollo de la aplicación. La planificación se realizó en Project Profesional 2015. Y está representado mediante un Diagrama de Gantt **Anexo 1. Plan de Actividades.**

Se estableció una duración de 110 días laborales con un total de 880 puntos estimados los cuales se cumplieron satisfactoriamente en las fechas propuestas.

3.1.2.1 Requerimientos

Son necesidades del cliente, los cuales definen el funcionamiento del sistema que se pretende desarrollar. Estos requerimientos deben ser recopilados del cliente como una lista llamada **Product Backlog** establecida por orden de prioridad.

Para la recopilación de estos requisitos se ha tenido 10 reuniones en la fundación amiga, con el presidente de la misma, tomando información básica y el proceso manual de control de asistencia de los empleados.

En esta etapa se definieron 38 Requisitos funcionales y 4 requisitos no funcionales, los mismos que serán documentados como historias de usuario, su código de representación será HU-01, donde el 01 es el número de la historia de usuario

Después de obtener la lista de historias de usuario general del proyecto se realizó el proceso de estimación y priorización para cada una de las historias.

En la **Tabla 7** se describe la Pila del Producto, en la que:

- ID es el identificador para cada una de las tareas presentadas.
- Tareas realizadas son las tareas establecidas para cada módulo del sistema.
- Estimación está dada por la relación tiempo-hombre, el tiempo está dado por horas tomando en cuenta que un día laborable es de 8 horas y el trabajo es realizado por una sola persona.

La columna “**Prioridad**” de la **Tabla 7** la determinamos según las necesidades de ejecución de la fundación, esta se describe en función de 3 parámetros Alta, Media y Baja, las cuales cuentan con un valor especificado en la **Tabla 7**. Este orden, se realizó en una reunión con el presidente de la fundación amiga en conjunto con el desarrollador del sistema, dándole una prioridad a cada tarea especificada en cada sprint.

Tabla 7-3 Prioridad de Ejecución

Prioridad	Valor
ALTA	3
MEDIA	2
BAJA	1

Realizado por: Erika R. 2017

Para determinar el esfuerzo que se necesita en el desarrollo de cada Historia Técnica o de Usuario se empleó el método T-Shirt, para ello se definió las tallas S, M, L y XL, la cual podemos visualizar en la Columna “**Talla**” de la **Tabla 8**, para efectuar esta técnica la secretaria de la fundación dio lectura a cada historia, se mencionó y se acordó el tiempo que se estima para realizar dicha historia, considerando la experiencia en proyectos similares.

A su vez que se estableció que 1 semana constará de 40 puntos estimados, de igual manera cada punto estimado semejará a una hora de trabajo. Se utilizó esta técnica de estimación por la facilidad de incrementos de forma cualitativa, además que permite hacer una estimación inicial rápida y fiable de esta manera las estimaciones son comprensibles para el Product Owner, y el desarrollador del sistema.

A continuación, mencionamos las tallas con su duración en la **Tabla 8**.

Tabla 8-3 Aplicación método T-Shirt

Talla	Puntos estimados	Horas de Trabajo
S	20	20
M	40	40
L	80	80
XL	160	160

Realizado por: Erika R. 2017

En el **Product Backlog** presentado (**Tabla 9**) se especifican los requerimientos del sistema, con la especificación antes mencionada.

Tabla 9-3 Product Backlog

ID	Requisitos	Prioridad	Puntos estimados	Talla
HT-01	Como desarrollador necesito realizar el estudio de la tecnología a utilizar.	3	20	S
HT-02	Como desarrollador necesito realizar el análisis y gestión de riesgos.	2	20	S
HT-03	Como desarrollador necesito realizar el diseño de la Base de Datos.	1	40	M
HT-04	Como desarrollador necesito realizar el diseño de la arquitectura del sistema.	3	20	S
HT-05	Como desarrollador necesito definir el estándar de programación.	2	20	S
HT-06	Como desarrollador necesito realizar el diagrama de clases del sistema.	1	20	S
HT-07	Como desarrollador necesito realizar el diseño de las interfaces.	1	20	S
HU-01	Como administrador necesito ingresar la información de empleados.	3	20	S
HU-02	Como administrador necesito actualizar la información de empleados.	2	20	S
HU-03	Como administrador necesito listar la información de empleados.	1	20	S
HU-04	Como administrador necesito eliminar la información de empleados.	1	20	S
HU-05	Como administrador necesito reingresar la información de empleados.	3	20	S
HU-06	Como administrador necesito actualizar el departamento de empleados de forma masiva.	2	20	S
HU-07	Como administrador necesito generar lista de cumpleaños del mes.	1	20	S
HU-08	Como administrador necesito enviar notificación de felicitación de cumpleaños a cada email.	1	20	S
HU-09	Como administrador necesito Ingresar los datos biométricos del empleado.	3	20	S
HU-10	Como administrador necesito actualizar biometrías de los empleados.	2	20	S
HU-11	Como administrador necesito listar biometrías de los empleados.	1	20	S

HU-12	Como administrador necesito generar el historial laboral del empleado.	1	20	S
HU-13	Como administrador necesito generar antigüedad laboral del empleado.	3	20	S
HU-14	Como administrador necesito actualizar el historial laboral del empleado.	2	20	S
HU-15	Como administrador necesito sincronizar los registros diarios del empleado.	1	20	S
HU-16	Como administrador necesito listar los registros diarios del empleado.	1	20	S
HU-17	Como administrador necesito generar los días de vacaciones a partir del registro diario	3	20	S
HU-18	Como administrador necesito generar asistencia a partir del registro diario del empleado.	2	20	S
HU-19	Como administrador necesito generar atrasos mediante el registro diario del empleado.	1	20	S
HU-20	Como administrador necesito actualizar del historial laboral por periodo del empleado.	1	20	S
HU-21	Como administrador necesito registrar solicitud de vacaciones del empleado.	3	20	S
HU-22	Como administrador necesito actualizar solicitud de vacaciones del empleado.	2	20	S
HU-23	Como administrador necesito anular solicitud de vacaciones del empleado	1	20	S
HU-24	Como administrador necesito listar las solicitudes de vacaciones por estado.	1	20	S
HU-25	Como administrador necesito generar días calendario laboral.	3	20	S
HU-26	Como administrador necesito actualizar días calendario laborable.	2	20	S
HU-27	Como administrador necesito listar días calendario laborable.	1	20	S
HU-28	Como administrador necesito registrar permisos por tipo.	1	20	S
HU-29	Como administrador necesito actualizar permisos registrados.	3	20	S
HU-30	Como administrador necesito listar permisos registrados.	2	20	S
HU-31	Como administrador necesito registrar horarios laborables.	1	40	M
HU-32	Como administrador necesito actualizar horarios laborables.	3	20	S
HU-33	Como administrador necesito eliminar horarios laborables.	2	20	S
HU-34	Como administrador necesito crear el reporte de empleados clasificados por departamento.	1	40	S
HU-35	Como administrador necesito crear el reporte de empleados que no están afiliados al IESS.			
HU-36	Como administrador necesito crear el reporte de asistencia por departamento.			
HU-37	Como administrador necesito crear el reporte de vacaciones gozadas, por gozar y saldos.			
HU-38	Como administrador necesito crear el reporte de vacaciones acumuladas.			

Realizado por: Erika R. 2017

3.1.2.2 Personas y roles del proyecto

Para la realización del sistema se contó con las siguientes personas que se detallan en la siguiente tabla.

Tabla 10-3 Personas y roles

Personas/Entidad	Rol
Fundación Amiga	Product Owner
Ing. Gloria Arcos	Scrum Master
Lcdo. Ricardo Batioja	Stake Holder
Erika Reina	Developer

Realizado por: Erika R. 2017

3.1.2.3 Tipos y roles de usuario

Durante el desarrollo del sistema se definieron 2 tipos de usuarios, los mismos que fueron establecidos a través de reuniones con los miembros de la “Fundación Amiga”, una vez definidos los requerimientos para los roles del usuario, la información obtenida se detalla en la siguiente tabla.

Tabla 11-3 Tipos y roles de usuarios

Tipo de Usuario	Rol
Administrador	<ul style="list-style-type: none">• Gestión de empleados• Generación de historial laboral• Generación de historial periodo• Generación de asistencia.• Generación de permisos• Generación de Atrasos• Generación de solicitud de vacaciones• Gestión de calendariol laboral• Generación de historial laboral diario• Obtención de reportes

Encargado de Recursos Humanos	<ul style="list-style-type: none"> • Aprobación de solicitud de vacaciones • Justificación de permisos • Justificación de atrasos • Visualización de registros diarios • Visualización de historial laboral • Visualización de asistencias • Obtención de reportes
-------------------------------	---

Realizado por: Erika R. 2017

3.1.2.4 Plan de Entrega

Es el resultado o producto de la planificación de los requerimientos del sistema. El proyecto se ejecuta en hitos transitorios pequeños y consolidados. Cada sprint tiene que proveer un entregable plenamente funcional del producto, de esta forma el cliente tendrá una versión del sistema cuando lo solicite, lo que se efectuará con un esfuerzo minúsculo.

Cada semana de trabajo se valora el trabajo desarrollado, se comunica cuáles son las dificultades presentadas y se actualiza el estado de la lista de tareas del Sprint Backlog.

Las tareas se organizan por sprints **tabla 12**; estos a su vez constan de 2 semanas de trabajo con una duración de 8 horas diarias; es decir 80 puntos estimados por sprint.

Tabla 12-3 Sprint Backlog

Sprint	ID	Requisitos	Total	Fecha Inicio	Fecha Fin
SPRINT 0	HT-01	Como desarrollador necesito realizar el estudio de la tecnología a utilizar.	80	jue 19/01/17	vie 20/01/17
	HT-02	Como desarrollador necesito realizar el análisis y gestión de riesgos.		lun 23/01/17	jue 26/01/17
	HT-03	Como desarrollador necesito realizar el diseño de la Base de Datos.		vie 27/01/17	mié 01/02/17
SPRINT 1	HT-04	Como desarrollador necesito realizar el diseño de la arquitectura del sistema.	80	jue 02/02/17	mar 07/02/17
	HT-05	Como desarrollador necesito definir el estándar de programación.		mié 08/02/17	jue 09/02/17
	HT-06	Como desarrollador necesito realizar el diagrama de clases del sistema.		vie 10/02/17	lun 13/02/17
	HT-07	Como desarrollador necesito realizar el diseño de las interfaces.		mar 14/02/17	mié 15/02/17
SPRINT 2	HU-01	Como administrador necesito ingresar la información de empleados.	80	mar 16/02/17	mié 21/02/17
	HU-02	Como administrador necesito actualizar la información de empleados.		jue 22/02/17	lun 23/02/17
	HU-03	Como administrador necesito listar la información de empleados.		vie 24/02/17	lun 27/02/17
	HU-04	Como administrador necesito eliminar la información de empleados.		mar 28/02/17	mié 01/03/17
	HU-05	Como administrador necesito reingresar la información de		jue	vie

SPRINT 3		empleados.	80	02/03/17	03/03/17
	HU-06	Como administrador necesito actualizar el departamento de empleados de forma masiva.		lun 06/03/17	mié 08/03/17
	HU-07	Como administrador necesito generar lista de cumpleaños del mes.		jue 09/03/17	lun 13/03/17
	HU-08	Como administrador necesito enviar notificación de felicitación de cumpleaños a cada email.		mar 14/03/17	mié 15/03/17
SPRINT 4	HU-09	Como administrador necesito Ingresar los datos biométricos del empleado.	80	jue 16/03/17	mié 22/03/17
	HU-10	Como administrador necesito actualizar biometrías de los empleados.		jue 23/03/17	vie 24/03/17
	HU-11	Como administrador necesito listar biometrías de los empleados.		lun 27/03/17	mar 28/03/17
	HU-12	Como administrador necesito generar el historial laboral del empleado.		mié 29/03/17	mié 29/03/17
SPRINT 5	HU-13	Como administrador necesito generar antigüedad laboral del empleado.	80	jue 30/03/17	lun 03/04/17
	HU-14	Como administrador necesito actualizar el historial laboral del empleado.		mar 04/04/17	jue 06/04/17
	HU-15	Como administrador necesito sincronizar los registros diarios del empleado.		vie 07/04/17	lun 10/04/17
	HU-16	Como administrador necesito listar los registros diarios del empleado.		mar 11/04/17	mié 12/04/17
SPRINT 6	HU-17	Como administrador necesito generar los días de vacaciones a partir del registro diario.	80	jue 13/04/17	lun 17/04/17
	HU-18	Como administrador necesito generar asistencia a partir del registro diario del empleado.		mar 18/04/17	jue 20/04/17
	HU-19	Como administrador necesito generar atrasos mediante el registro diario del empleado.		vie 21/04/17	mar 25/04/17
	HU-20	Como administrador necesito actualizar del historial laboral por periodo del empleado.		mié 26/04/17	mié 26/04/17
SPRINT 7	HU-21	Como administrador necesito registrar solicitud de vacaciones del empleado.	80	jue 27/04/17	vie 28/04/17
	HU-22	Como administrador necesito actualizar solicitud de vacaciones del empleado.		lun 01/05/17	mar 02/05/17
	HU-23	Como administrador necesito anular solicitud de vacaciones del empleado.		mié 03/05/17	mar 09/05/17
	HU-24	Como administrador necesito listar las solicitudes de vacaciones por estado.		mié 10/05/17	mié 10/05/17
SPRINT 8	HU-25	Como administrador necesito generar días calendario laboral.	80	jue 11/05/17	lun 15/05/17
	HU-26	Como administrador necesito actualizar días calendario laborable.		mar 16/05/17	mié 17/05/17
	HU-27	Como administrador necesito listar días calendario laborable.		jue 18/05/17	lun 22/05/17
	HU-28	Como administrador necesito registrar permisos por tipo.		mar 23/05/17	mié 24/05/17
SPRINT 9	HU-29	Como administrador necesito actualizar permisos registrados.	80	jue 25/05/17	vie 26/05/17
	HU-30	Como administrador necesito listar permisos registrados.		lun 29/05/17	mar 30/05/17
	HU-31	Como administrador necesito registrar horarios laborables.		vie 31/05/17	mié 07/06/17
	HU-32	Como administrador necesito actualizar horarios laborables.		jue 08/06/17	vie 09/06/17
	HU-33	Como administrador necesito eliminar horarios laborables.		lun 12/06/17	mar 13/06/17
	HU-34	Como administrador necesito crear el reporte de trabajadores clasificados por departamento.		mié 14/06/17	jue 15/06/17

SPRINT 10		.	80		
	HU-35	Como administrador necesito crear el reporte de quienes no están afiliados al IESS.		vie 16/06/17	vie 16/06/17
	HU-36	Como administrador necesito crear el reporte de asistencia por departamento.		lun 19/06/17	lun 19/06/17
	HU-37	Como administrador necesito crear el reporte de vacaciones gozadas, por gozar y saldos.		mar 20/06/17	mar 20/06/17
	HU-38	Como administrador necesito crear el reporte de vacaciones acumuladas.	mié 21/06/17	mié 21/06/17	
		TOTAL	880		

Realizado por: Erika R. 2017

Se establecieron 880 puntos en el plan de entrega los cuales se cumplieron a cabalidad.

3.1.3 Fase de Desarrollo de Sprints

3.1.3.1 Sprint 0

3.1.3.1.1 Tecnología utilizada

Se seleccionó las herramientas que ayudaron en el desarrollo del sistema, debido a que en la Fundación Amiga tienen otras aplicaciones informáticas en las cuales se emplean estas herramientas, esto se determinó en 5 reuniones con el presidente de la Fundación y el jefe del departamento técnico por lo que se designó lo siguiente:

Se ha implementado el sistema utilizando el IDE Netbeans 8.1, el JDK 1.8 bajo el lenguaje de programación java, framework jsf, hibernate para el mapeo de los datos, el servidor GlassFish 4.1.1. y PostgreSQL 9.5 como gestor de Base de Datos, además de PgAdmin 1.16.1 para gestionar el gestor de bases de datos.

Para el modelado de los diagramas bases del sistema se dejó a elección del desarrollador. Se empleó las siguientes herramientas de diseño debido a que se contaba con conocimientos sólidos del manejo de dichas aplicaciones lo que permitió facilidad y eficiencia en el modelado de las mismas, estas apoyaron en la gestión del proyecto como son: PowerDesigner 16.5, StarUml 5.0.2.1 y Visio 2016, que se irán mencionando a medida que se las haya empleado en el proyecto. Para el desarrollo de la documentación se utilizó la herramienta Microsoft Word 2016.

De igual manera para la implementación del requerimiento técnico mencionado anteriormente se ha realizado su respectiva historia técnica, tareas de ingeniería y pruebas de aceptación las cuales se muestran a continuación.

Tabla 13-3 Estudio de la Tecnología a utilizar

HISTORIA TÉCNICA	
Número: HT-01	Nombre Historia Técnica: Estudio de la Tecnología a utilizar
Usuario: Desarrollador	Sprint Asignado: 00
Fecha Inicio: 19/01/17	Fecha Fin: 20/01/17
Descripción: Como desarrolladores del sistema necesitamos analizar la tecnología que se empleará en el mismo.	
Pruebas de aceptación: La tecnología a utilizar satisface las necesidades del sistema.	

Realizado por: Erika R. 2017

Tabla 14-3 Prueba de aceptación 1 para HT-01

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Historia Técnica: Estudio de la Tecnología a utilizar
Nombre de la Prueba: La tecnología a utilizar satisface las necesidades del sistema.	
Responsable: Erika Reina	Fecha: 20/01/17
Descripción: Se deberá analizar si la tecnología que se va a utilizar cumple con las necesidades de desarrollo.	
Condiciones de Ejecución: Tener claro las necesidades del proyecto.	
Pasos de Ejecución <ul style="list-style-type: none"> • Investigar herramientas de desarrollo. • Verificar que satisfagan las necesidades del sistema. 	
Resultado Esperado: <ul style="list-style-type: none"> • Tecnología adecuada. 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

Las tareas de ingeniería realizada en la HT-01 se definen en la siguiente tabla

Tabla 15-3 Tabla de Actividades para HT-02

FECHA	ACTIVIDAD	TIEMPO
19/01/17 -20/01/17	<p>Tarea 1: Determinación de la Tecnología a utilizar</p> <ul style="list-style-type: none"> • Se seleccionará de acuerdo a las necesidades del proyecto las herramientas de desarrollo. 	16 horas

Realizado por: Erika R. 2017

Tabla 16-3 Tarea de Ingeniería 1 para la HT-02

TAREA DE INGENIERÍA	
SPRINT: 00	Número de Tarea: 01
Nombre Historia Técnica: Estudio de la Tecnología a utilizar	
Nombre de la Tarea: Determinación de la Tecnología a utilizar	
Tipo de Tarea: Análisis	Programador Responsable: Erika Reina
Fecha de Inicio: 19/01/17	Fecha Fin: 20/01/17
Descripción: Se seleccionará de acuerdo a las necesidades del proyecto las herramientas de desarrollo.	
Pruebas de Aceptación: Verificar que las herramientas seleccionadas permitan dar una solución óptima al proyecto.	

Realizado por: Erika R. 2017

Tabla 17-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Definición de los requerimientos
Nombre de la Prueba: Verificar que las herramientas seleccionadas permitan dar una solución óptima al proyecto.	
Responsable: Erika Reina	Fecha: 20/01/17
Descripción: Se verificará que las herramientas que se escogieron permitan dar una solución efectiva al proyecto.	
Condiciones de Ejecución: Tener conocimiento de herramientas de desarrollo de software.	
Pasos de Ejecución:	
<ul style="list-style-type: none"> • Analizar las herramientas seleccionadas 	
Resultado Esperado: Se seleccionaron correctamente las herramientas de desarrollo.	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

3.1.3.1.2 *Análisis y Gestión de Riesgos*

Análisis de Riesgos

Esto se realizó ya que pueden existir posibles inconvenientes a la hora de llevar a cabo un proyecto determinado, esto se debe tener en cuenta para realizar una debida acción que cubra con aquella deficiencia evitando el fracaso del proyecto.

Un riesgo puede generar cambios de opinión, acciones, etc. Debido a que esto no se puede evitar ocasiona incertidumbre y perdidas no deseadas cuando el riesgo se transforma en un problema.

Para gestionar los riesgos se debe tomar en cuenta el tipo, la prioridad y lo que podría ocasionar, en caso de efectuarse.

Identificación del Riesgo

Se definieron 5 riesgos que pudieron ser amenazas dentro del proyecto. Para ello se tomó en cuenta tres tipos de riesgos: Riesgo Técnico, Riesgo del Negocio y Riesgo del Proyecto. A continuación, se detalla lo mencionado.

Tabla 18-3: Identificación del Riesgo

ID	DESCRIPCIÓN DEL RIESGO	TIPO	CONSECUENCIA
R1	Cancelación de actividades del Negocio.	Negocio	Fracaso del Proyecto.
R2	Pérdida o robo de los equipos de trabajo	Proyecto	Pérdida de información. Pérdidas económicas Pérdidas de tiempo. Retraso en la entrega de proyecto.
R3	Falla en los servidores	Técnico	Perdidas económicas Pérdida de la información Errores en las transacciones.
R4	Cambio de políticas de la empresa que incurren en las metas y objetivos del proyecto.	Proyecto	El proyecto puede sufrir retrasos, cuanto más avanzado este el desarrollo del mismo más crítico será implementar los cambios, y generación de nuevos requisitos.
R5	El usuario cambio constantemente los requerimientos propuestos	Proyecto	Rediseño del Proyecto Re-planificación del Proyecto. Incremento de costos. Retraso del proyecto

Realizado por: Erika R. 2017

Probabilidad de Riesgos

Fue necesario determinar la probabilidad que generaría el riesgo de acuerdo a su complejidad. Como se presenta a continuación.

Tabla 19-3: Probabilidad del Riesgo

ID	PROBABILIDAD
R1	10
R2	15
R3	50
R4	20
R5	70

Realizado por: Erika R. 2017

Categorización del Riesgo

La variable considerada para este punto es el impacto negativo que puede tener un riesgo en cuanto al correcto desarrollo del proyecto. Por esto se enfocó en aquellos riesgos que afecten directamente al avance del trabajo.

Valoración de la Probabilidad

Considerando la prioridad y el impacto antes mencionado se estableció el rango de probabilidad que tendría cada riesgo.

Tabla 20-3: Probabilidad del Riesgo

Rango de Probabilidad	Descripción	Valor
1% - 33%	Baja	1
34% - 67%	Media	2
68% - 99%	Alta	3

Realizado por: Erika R. 2017

Valoración del Impacto

El impacto determinará el tiempo de retraso que tendrá el proyecto en caso de efectuarse un determinado riesgo.

Tabla 21-3: Valoración del Impacto

Impacto	Retraso	Impacto Técnico	Valor
Bajo	1 semana	Ligero impacto en el desarrollo del proyecto	1
Moderado	2 semanas	Moderado efecto en el desarrollo del proyecto	2
Alto	1 mes	Severo efecto en el desarrollo del proyecto	3
Crítico	Más de 1 mes	Proyecto no puede ser culminado	4

Realizado por: Erika R. 2017

Valoración de la exposición del riesgo

Mediante una categorización de colores se podrá determinar de mejor maneja el valor que tendrá cada uno de los riesgos, siendo estos colores verde, amarillo y rojo.

Tabla 22-3 Valoración de la exposición del riesgo

EXPOSICIÓN DEL RIESGO	VALOR	COLOR
Baja	1 o 2	Verde
Media	3 o 4	Amarillo
Alta	> 4	Rojo

Realizado por: Erika R. 2017

Tabla 23-3 Exposición del riesgo

Prob./Impacto.	Bajo=1	Moderado=2	Alto=3	Crítico=4
Alta=3	3	6	9	12
Media=2	2	4	6	8
Baja=1	1	2	3	4

Realizado por: Erika R. 2017

Determinación de la probabilidad del riesgo

A continuación, se presenta el análisis de los puntos antes mencionados en conjunto con cada uno de los riesgos expuestos para este proyecto.

Tabla 24-3: Resumen riesgos

Identificación	Probabilidad			Impacto		Exposición al riesgo	
	%	Valor	Probabilidad	Valor	Impacto	Valor	Expo
R1	10	1	Baja	4	Crítico	4	Media
R2	15	1	Baja	3	Alto	3	Media
R3	50	2	Media	2	Moderado	4	Media
R4	20	1	Baja	3	Alto	3	Media
R5	70	3	Alta	3	Alto	9	Alta

Realizado por: Erika R. 2017

Gestión de Riesgos

En el **Anexo 3**, se puede visualizar las hojas de gestión de riesgos esto es un plan de reducción, supervisión y gestión de cada uno de los riesgos presentados.

3.1.3.1.3 *Diseño de la base de datos*

Con el fin de prolongar la persistencia de la información y datos, manejados y almacenados por la “Fundación Amiga”, así como obtener acceso a información exacta y actualizada, se realizó el diseño de la base de datos mediante un Diagrama Entidad Relación (DER) **Figura 8-1**, utilizando la herramienta Visio 2016, este diseño permite identificar las entidades involucradas en el desarrollo del sistema, además entender las relaciones que sean relevantes entre las diversas entidades, describiendo el tipo de relación que existe.

Existen **5 entidades** las cuales son: Empleado, Biometría, Registro, Historial_Laboral y Dispositivo.

Los atributos para cada entidad se encuentran descritos en el modelo detallado en el **gráfico 6-3**, su identificación se escribe todo en minúscula y en caso de que sean palabras compuestas se separan con un “_” y son los que se han considerado necesarios, para almacenar en la base de datos.

A continuación, se describe las relaciones entre entidades del Diagrama Entidad Relación del sistema “SISTAHU”:

- Existe un empleado pertenece a un departamento.
- A su vez un empleado tendrá un historial laboral (contará con atrasos, permisos, etc)
- El empleado posee una biometría (huellas dactilares únicas).
- Esta biometría va a generar varios registros a lo largo de un periodo.
- Estos registros se grabarán en un dispositivo (lector de huellas digitales).

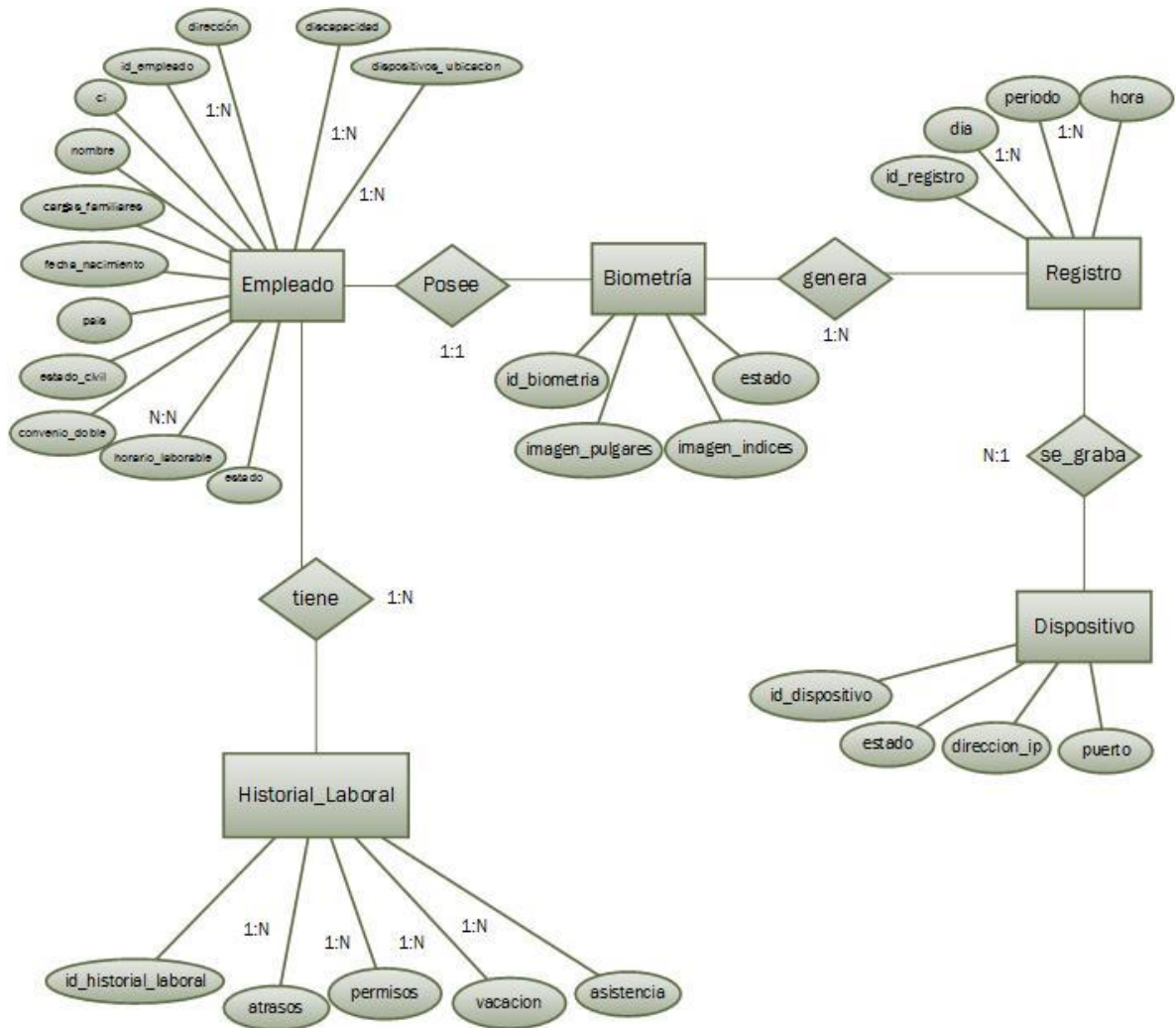


Gráfico 6-3: Diagrama Entidad Relación
 Realizado por: Erika R. 2017

- **Normalización de los datos**

Con el objetivo de eliminar redundancias e inconsistencias de dependencia en el diseño de las tablas se realizó un análisis de los datos del diagrama entidad relación; tablas y atributos, para que se puedan acomodar a futuros cambios en el negocio y a su vez minimizar el impacto que estos tendrían en el sistema, de esta manera se guarde la información de manera eficaz, fiable, flexible y apropiada.

Para ello se realizó la respectiva normalización en donde se aplicó la 1ra, 2da y 3ra forma normal. La primera forma normal se aplicó para que cada campo de una fila contenga sólo un valor; es decir que prohíbe a un campo contener múltiples valores, además establece que exista un identificador

para cada tabla, lo que garantiza **unicidad e integridad de los datos**. La 2da forma normal se aplicó para que no exista **redundancia de los datos** en una tabla, lo que garantiza la inserción de un registro sin un exceso de datos en las tablas. La 3ra forma normal se aplicó para eliminar la dependencia transitiva evitando errores de lógica cuando se insertan o borran registros, lo que garantiza facilidad de trabajo y expansión.

Como resultado a esto se realizó el Diagrama Lógico de la Base de datos **gráfico 7-3**, el cual está compuesto por **17 tablas** obtenidas del proceso de normalización antes mencionado, se empleó la herramienta Power Designer 16.5 para su creación.

Se llegó sólo hasta la 3ra forma normal ya que se consideró necesario para darle solución al problema planteado, lo que garantizará la simplificación de una estructura de datos compleja, optimizando el consumo de espacio de la base de datos.

- **Implementación de la base de datos**

Se realizó con el objetivo de implementar de forma eficiente en las estructuras físicas del Sistema de Gestión de Base de Datos seleccionado el Diagrama Lógico, para optimizar el desempeño del sistema.

El esquema físico de la base de datos fue generado a través del refinamiento del Diagrama Lógico y este a través del Sistema de Gestión de Base de datos PostgreSQL, a partir de la herramienta Power Designer 16.5 mediante una conexión con el mismo.

El nombre de la base de datos es: **SISTAHU**, donde se creó un esquema de nombre **fundacion** el cual va a contener las 17 tablas establecidas en el diagrama lógico con sus respectivos atributos.

El esquema físico de la base de datos se encuentra desplegado en el Sistema de Gestión de Base de datos PostgreSQL 9.5 instalado en una máquina con Sistema Operativo CentOS 7.

- **Documentación**

Se realizó con el objetivo de presentar la información básica de las tablas que componen la base de datos; es decir atributos y características de los mismos, para ello se creó un diccionario de datos el cual es una colección de datos que componen el flujo de los mismos, en la cual se describen sus propiedades, el diccionario de datos de la base **SISTAHU** se generó a través de un código específico **figura 10-3.** para postgresQL, la cual se encuentra organizada por tablas alfabéticamente, describiendo cada columna de la tabla.

Entre las propiedades que describen las columnas se encuentran: El nombre de la tabla, nombre de la columna, el tipo de dato y longitud de la columna.

Los tipos de datos de las columnas de acuerdo al SGBD seleccionado queda definido de la siguiente manera:

Código del registro: *serial*

Cantidades enteras: *int*

Cadena de texto: *varchar*

Fechas: *date*

Horas: *time*

Imágenes: *byte*

Valores de Verdadero y Falso: *boolean*

Todas las tablas cuentan con el parámetro *es_activo* que sirve como un marcador para cada registro. Por ejemplo, se marca los empleados activos TRUE y cuando se da de baja a uno, automáticamente cambia el estado del empleado a FALSE.

El diccionario de datos se encuentra en el **Anexo 2**. El cual es muy importante para el mantenimiento de la base de datos.

```
isc.column_default::character varying AS column_default,  
isc.data_type::character varying AS data_type,  
isc.character_maximum_length::integer AS str_length,  
    CASE  
        WHEN isc.udt_name::text = 'int4'::text OR isc.udt_name::text =  
bool'::text THEN isc.data_type::character varying  
        ELSE isc.udt_name::character varying  
    END AS udt_name  
  
FROM information_schema.columns isc  
WHERE isc.table_schema::text = 'fundacion'::text  
ORDER BY isc.table_name, isc.ordinal_position;
```

Figura 10-3: Código para generar diccionario de datos

Realizado por: (Emmanuel, 2015)

Para la implementación de este requerimiento no funcional se ha realizado su respectiva historia técnica, tareas de ingeniería y pruebas de aceptación las cuales están desarrolladas a continuación.

Tabla 25-3: HT-03 Diseño de la base de datos.

HISTORIA TÉCNICA	
Número: HT-03	Nombre Historia Técnica: Diseño de la base de datos.
Usuario: Desarrollador	Sprint Asignado: 01
Fecha Inicio: 27/01/17	Fecha Fin: 01/02/17
<p>Descripción: Como desarrollador necesito construir la base de datos del sistema para poder tener acceso a los mismos, que se pueden visualizar, ingresar o actualizar, en concordancia con los permisos que se les hayan asignado a los usuarios.</p>	
<p>Pruebas de aceptación:</p> <ul style="list-style-type: none"> • Se encuentre correctamente estructurada la base de datos. 	

Realizado por: Erika R. 2017

Tabla 26-3: Prueba de aceptación 1 para HT-03

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Historia Técnica: Diseño de la base de datos.
<p>Nombre de la Prueba: Se encuentre correctamente estructurada la base de datos.</p>	
Responsable: Erika Reina	Fecha: 07/02/17
<p>Descripción: Se verificará que el modelo de base de datos cumpla con un diseño correcto evitando la redundancia de datos.</p>	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Tener claro los requerimientos del sistema. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Verificar que la base de datos creada cumpla con el modelo aprobado por los interesados del proyecto. 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • La base de datos posee una estructura correcta. 	
<p>Evaluación de la Prueba: Satisfactoria</p>	

Realizado por: Erika R. 2017

Las tareas de ingeniería realizada en la HT-03 se definen en la siguiente tabla

Tabla 27-3: Tabla de Actividades para HT-03

FECHA	ACTIVIDAD	TIEMPO
27/01/17 – 27/01/17	Tarea 1: Diseño del Modelo Entidad Relación <ul style="list-style-type: none"> • Análisis de los procesos • Identificar entidades y relaciones 	8 horas
30/01/17 - 30/01/17	Tarea 2: Creación del diagrama Lógico <ul style="list-style-type: none"> • Normalizar las tablas. • Crear el diagrama lógico. 	8 horas
31/01/17 - 31/01/17	Tarea 3: Creación del modelo Físico <ul style="list-style-type: none"> • Generar el Script a ejecutarse en nuestro gestor de base de dalos • Generar el Diccionario de Datos 	8 horas
01/03/17 - 01/03/17	Tarea 4: Creación de la base de datos en el DBMS PostgreSQL 9 <ul style="list-style-type: none"> • Crear la base de datos • Crear el esquema de base de Datos. 	8 horas

Realizado por: Erika R. 2017

Tabla 28-3: Tarea de Ingeniería 1 para la HT-05

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 01
Nombre Historia Técnica: Diseño de la base de datos.	
Nombre de la Tarea: Diseño del Modelo Entidad Relación.	
Tipo de Tarea: Análisis	Programador Responsable: Erika Reina
Fecha de Inicio: 27/01/17	Fecha Fin: 27/01/17
Descripción <ul style="list-style-type: none"> • Análisis de los procesos • Identificar entidades y relaciones 	

Pruebas de Aceptación:

- Aprobación del Jefe de Sistemas de la Fundación Amiga el diseño de la base de datos

Realizado por: Erika R. 2017

Tabla 29-3: Prueba de aceptación 1 para tara de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Diseño del Modelo Entidad Relación.
Nombre de la Prueba: Aprobación del Jefe de Sistemas de la Fundación Amiga el diseño de la base de datos.	
Responsable: Erika Reina	Fecha: 27/01/17
Descripción: Se constatará que el Jefe de Sistemas de la Fundación esté de acuerdo con el diseño de la base de datos.	
Condiciones de Ejecución	
Pasos de Ejecución <ul style="list-style-type: none">• Verificar la no redundancia de datos.• Verificar la no dependía de claves primaras.• Realizar el Diseño de la base de datos.	
Resultado Esperado: <ul style="list-style-type: none">• Aprobación del diseño de la base de datos.	
Evaluación de la Prueba: <ul style="list-style-type: none">• Satisfactoria con leves cambios realizados.	

Realizado por: Erika R. 2017

Tabla 30 -3: Tarea de Ingeniería 2 para la HT-03

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 02
Nombre Historia Técnica: Diseño de la base de datos.	
Nombre de la Tarea: Creación del diagrama Lógico	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina
Fecha de Inicio: 30/01/17	Fecha Fin: 30/01/17
Descripción <ul style="list-style-type: none"> • Normalizar las tablas. • Crear el diagrama lógico. 	
Pruebas de Aceptación: Verificar que el diagrama de base de datos aprobado sea implementado en Power Designer 16.5.	

Realizado por: Erika R. 2017

Tabla 31-3: Prueba de aceptación 1 para tara de ingeniería 2

PRUEBAS DE ACEPTACIÓN	
Código: 2.1	Nombre Tarea de Ingeniería: Creación del diagrama Lógico
Nombre de la Prueba: Verificar que el diagrama de base de datos aprobado sea implementado en Power Designer 16.5.	
Responsable: Erika Reina	Fecha: 30/01/17
Descripción: Se realizará el diagrama lógico de la base de datos aprobado, en el software seleccionado	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Tener instalado el software de modelado de base de datos Power Designer 16.5 	
Pasos de Ejecución <ul style="list-style-type: none"> • Verificar que se desarrolle en Power Designer 16.5 el modelo aprobado de la Base de Datos. 	
Resultado Esperado: <ul style="list-style-type: none"> • El diagrama realizado en Power Designer coincide con el modelo de Base de Datos aprobado. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Erika R. 2017

Tabla 32-3: Tarea de Ingeniería 3 para la HT-05

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 03
Nombre Historia Técnica: Diseño de la base de datos.	
Nombre de la Tarea: Creación del modelo Físico	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina
Fecha de Inicio: 31/01/17	Fecha Fin: 31/01/17
Descripción <ul style="list-style-type: none"> • Generar el Script a ejecutarse en nuestro gestor de base de dalos • Generar el Diccionario de Datos 	
Pruebas de Aceptación: Verificar que el script de base de datos y el diccionario del mismo se generen de acuerdo al modelo lógico aprobado.	

Realizado por: Erika R. 2017

Tabla 33-3: Prueba de aceptación 1 para tara de ingeniería 3

PRUEBAS DE ACEPTACIÓN	
Código: 3.1	Nombre Tarea de Ingeniería: Creación del modelo Físico.
Nombre de la Prueba: Verificar que el script de base de datos y el diccionario del mismo se generen de acuerdo al modelo lógico aprobado.	
Responsable: Erika Reina	Fecha: 30/01/17
Descripción: Se verificará que el script de base de datos y el diccionario del mismo se generen correctamente de acuerdo a los lineamientos establecidos modelo lógico aprobado.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Tener creado el modelo lógico de la Base de Datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Crear en Script de acuerdo al modelo lógico aprobado • Verificar que el script este correctamente generado • Crear el Diccionario de Datos en Power Designer 	
Resultado Esperado: <ul style="list-style-type: none"> • El script y el diccionario de datos se generaron correctamente 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Erika R. 2017

Tabla 34-3: Tarea de Ingeniería 4 para la HT-05

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 04
Nombre de Historia Técnica: Diseño de la base de datos.	
Nombre de la Tarea: Creación de la base de datos en el SGBD PostgreSQL 9.5	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina
Fecha de Inicio: 01/02/17	Fecha Fin: 01/02/17
Descripción <ul style="list-style-type: none"> • Crear la base de datos. • Crear el esquema de base de Datos. 	
Pruebas de Aceptación: Verificar que se haya creado correctamente la base de datos en el DBMS.	

Realizado por: Erika R. 2017

Tabla 35-3: Prueba de aceptación 1 para tara de ingeniería 4

PRUEBAS DE ACEPTACIÓN	
Código: 4.1	Nombre Historia Técnica: Creación de la base de datos en el SGBD PostgreSQL 9.5
Nombre de la Prueba: Verificar que se haya creado correctamente la base de datos en el SGBD.	
Responsable: Erika Reina	Fecha: 01/02/17
Descripción: Se verificará en el SGBD que se haya creado correctamente la base de datos.	
Condiciones de Ejecución <ul style="list-style-type: none"> • Que haya conexión con el servidor 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ejecutar el script creado en el DBMS. 	
Resultado Esperado: <ul style="list-style-type: none"> • Se creó correctamente la base de Datos en el DBMS 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria. 	

Realizado por: Erika R. 2017

3.1.3.2 Sprint 1

3.1.3.2.1 Diseño de la arquitectura del sistema

Realizar la representación de la arquitectura del sistema permitirá modelar el mismo, así como cada uno de los componentes que lo constituyen y cómo se desplegarán. El sistema SISTAHU consta de una arquitectura Cliente Servidor debido a que las tareas se reparten entre los servidores y los demandantes clientes. El cliente; es decir un pc realizará peticiones al servidor que será quien emita la respuesta.

Para la implementación de la arquitectura se ha realizado el diagrama de despliegue y el diagrama de componentes del sistema mediante la herramienta StarUML.

El diagrama de despliegue **figura 11-3** consta de un lector de huellas digitales que se conectará directamente al SISTAHU, 2 servidores con Windows server 2012 representados mediante 2 Nodos en el que estarán almacenados el sistema gestor de base de datos en el servidor de Base de datos y la aplicación SISTAHU en el servidor web GlashFish 4.1, así como un switch TP-Link representado con otro nodo el cual permitirá la comunicación entre los servidores a través de Internet mediante una red Local.

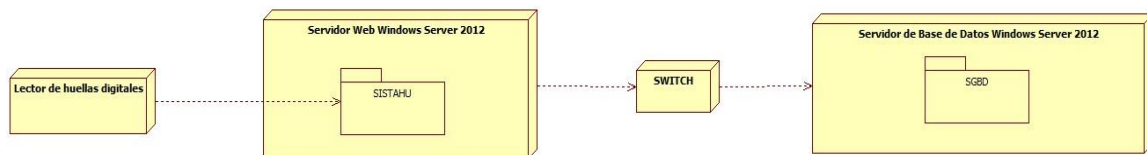


Figura 11-3: Diagrama de despliegue

Realizado por: Erika R. 2017

En la **figura 12-3**. Se describe mediante un diagrama de componentes; el componente Sistahu que contiene la aplicación la cual está dividida en 3 capas: la capa de acceso a datos en la cual se accederá a los mismos mediante Hibernate, la capa de negocio en la que se trabajará con sentencias HQL propias de Hibernate las cuales se encontrarán establecidas en archivos .jar y la capa de presentación en donde se utilizó Primefaces el cual definió el diseño MVC; el modelo en donde se trabajará a través de entidades, la vista en donde se empleará el uso de primfaces 5.3 y el controlador en el que se utilizará el framework JSF 2.2 y el componente SGBD que contendrá la base de datos PostgreSQL 9.5.

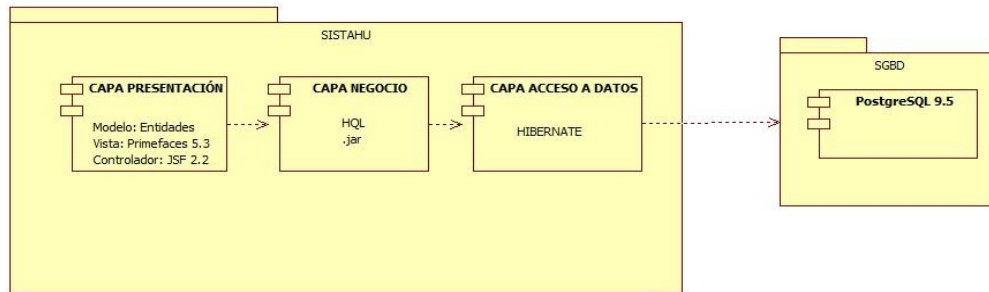


Figura 12-3: Diagrama de Componentes
Realizado por: Erika R. 2017

Posterior a reuniones con el Stake Holder y los interesados del proyecto se acordó utilizar la arquitectura en 3 capas. Para la implementación de este requerimiento no funcional se ha realizado su respectiva historia técnica, tareas de ingeniería y pruebas de aceptación las cuales están desarrolladas a continuación.

Tabla 36-3: HT-04 Diseño de la arquitectura del sistema

HISTORIA TÉCNICA	
Número: HT-04	Nombre Historia Técnica: Diseño de la arquitectura del sistema.
Usuario: Desarrollador	Sprint Asignado: 01
Fecha Inicio: 02/02/17	Fecha Fin: 07/02/17
Descripción: Como desarrollador necesito determinar la arquitectura del sistema de acuerdo a la petición del usuario y los recursos con los que cuenta la Fundación Amiga.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> En el diagrama UML verificar cada componente de la arquitectura. 	

Realizado por: Erika R. 2017

Tabla 37-3: Prueba de aceptación 1 para HT-04

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Historia Técnica: Como desarrollador necesito determinar la arquitectura del sistema.
Nombre de la Prueba: En el diagrama UML verificar cada componente de la arquitectura.	
Responsable: Erika Reina	Fecha: 07/02/17
Descripción: En un diagrama UML verificar que cada componente de la arquitectura permita cumplir satisfactoriamente la funcionalidad del sistema.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> Los dispositivos identificados deben estar disponibles. 	
Pasos de Ejecución	
<ul style="list-style-type: none"> Verificar si los elementos identificados pueden relacionarse satisfactoriamente de acuerdo a la necesidad del sistema. 	

Resultado Esperado: <ul style="list-style-type: none"> El diagrama de despliegue refleja una arquitectura que se puede desplegar en la realidad.
Evaluación de la Prueba: Satisfactoria

Realizado por: Erika R. 2017

Las tareas de ingeniería realizada en la HT-06 se definen en la siguiente tabla:

Tabla 38-3: Tabla de Actividades para HT-04

FECHA	ACTIVIDAD	TIEMPO
02/02/17- 07/02/17	Tarea 1: Definición de la arquitectura del sistema	32 horas

Realizado por: Erika R. 2017

Tabla 39-3: Tarea de Ingeniería 1 para la HT-04

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 01
Nombre Historia Técnica: Diseño de la arquitectura del sistema.	
Nombre de la Tarea: Definición de la arquitectura del sistema	
Tipo de Tarea: Análisis	Programador Responsable: Erika Reina
Fecha de Inicio: 02/02/17	Fecha Fin: 07/02/17
Descripción: Definición de la arquitectura del sistema	
Pruebas de Aceptación <ul style="list-style-type: none"> Aprobación de la arquitectura del sistema por parte del Jefe de Sistemas de la Fundación Amiga. 	

Realizado por: Erika R. 2017

Tabla 40-3: Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Definición de la arquitectura del sistema.
Nombre de la Prueba: Aprobación de la arquitectura del sistema por parte del Jefe de Sistemas de la Fundación Amiga.	
Responsable: Erika Reina	Fecha: 07/02/17
Descripción: Determinar correctamente cada una de las capas de la arquitectura, para el modelo, las vistas y el controlador respectivamente.	
Condiciones de Ejecución: Tener clara la idea del modo de funcionamiento del patrón de MVC	
Pasos de Ejecución	

<ul style="list-style-type: none"> • Investigar sobre el patrón de arquitectura MVC. • Clasificar adecuadamente el modelo, las vistas y el controlador para el sistema. • Realizar el despliegue de los componentes.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Aprobación de la definición de la arquitectura.
<p>Evaluación de la Prueba:</p> <p>Satisfactoria</p>

Realizado por: Erika R. 2017

3.1.3.2.2 Definición del estándar de codificación

Implementar un estándar de codificación permitirá tener una estructura detallada de cómo manejar la sintaxis del código de la aplicación; es decir establecer una norma de cómo escribir el código en el proyecto lo que permitirá que las nuevas implementaciones en la aplicación puedan ser llevadas a cabo con facilidad ya que será de fácil entendimiento para los desarrolladores. (Administrador, 2014)

El estándar de codificación utilizado en este proyecto es el Estándar de Java, ya que al utilizar la combinación de UpperCamelCase y lowerCamelCase se obtiene un mejor entendimiento de la estructura del sistema lo que permite de una manera sencilla y fácil recordar cada uno de los atributos, métodos, paquetes, etc. que forman parte de nuestra codificación del sistema. (Administrador, 2014)

Para la base de datos se utilizó el nombre de la misma con mayúsculas, todos los nombres tanto de tablas, campos y esquema se definieron con minúsculas separando cada palabra con “_”, debido a que Hibernate es sensible a las mayúsculas o minúsculas, es decir no soporta el uso de mayúsculas en los campos para el mapeo de datos. (Los nombres de las clases Mapeadas por Hibernate se definen con UpperCamelCase). (Administrador, 2014)

El análisis y aprendizaje del estándar para normalizar el código, puede ser aplicado en el proyecto y posteriormente en nuevas reformas que se realicen facilitando su entendimiento, ya sea para: nombres de funciones, métodos, procedimientos, variables, vistas, nombre de atributos en la base de datos, por lo que los interesados del proyecto aprueban el estándar de codificación.

A continuación, se presenta la descripción del estándar seleccionado (Administrador, 2014):

Nomenclatura

Se utilizará una combinación en inglés y el nombre funcional para denominar a las clases, variables, constantes, etc. La parte funcional se escribirá en castellano, por ejemplo: getListEmpleados. (Administrador, 2014)

Paquetes

Cada uno de los paquetes se determinará en minúsculas evitando la utilización de caracteres especiales. (Administrador, 2014)

Nombres de Interfaces

En cuanto a los nombres de las interfaces se empleará el sufijo Interface en donde se añadirán palabras con inicial en mayúscula. Debe evitarse las abreviaciones para hacer más simple de entender el código. Ejemplo: EmpleadoInterface. (Administrador, 2014)

Nombres de clases

Los nombres de las clases deben comenzar con mayúscula, las palabras deben ser completas evitando el uso de acrónimos y abreviaturas. Los permitidos son: DAO, DTO, URL, HTML, etc. (Administrador, 2014)

Nombres específicos de gestiones

Para realizar la gestión de una entidad específica (Ej. Empleado) se definen con las reglas a continuación: Clase: <<FuncionalidadGenerica>><<Entidad>><<Especificación de Clase>>. (Administrador, 2014)

Métodos

En cuanto a los métodos se deben usar verbos en infinitivo con la inicial minúscula y la inicial de cada palabra interna con mayúscula. Sin usar caracteres especiales. (Administrador, 2014)

Variables

Los nombres de las variables tanto de instancia como estáticas reciben el mismo tratamiento que para los métodos, con la salvedad de que aquí sí importa más la relación entre la regla mnemónica y la longitud del nombre. Se escribirán de la misma manera que los métodos, sin embargo, se debe

tomar en cuenta que es más importante la relación entre la regla mnemónica, así como la longitud del nombre. (Administrador, 2014)

Constantes

Serán escritas en mayúsculas separando las palabras por un “_”. (Administrador, 2014)

Estilo de codificación

Comentarios

Estos deben ser claros y concisos brindando información sobre la implementación del diseño de dicha clase, evitando caracteres especiales, cajas u otros gráficos creados mediante código ASCII. (Administrador, 2014)

Comentarios de Documentación (JavaDoc)

Se debe definir un comentario de documentación por cada clase o interface, método, propiedad o constante creado. Se debe comentar las variables de tipo private o protected. (Administrador, 2014)

Declaraciones

La declaración de las variables debe ser una de cada vez. Además, deben inicializarse todas las variables locales salvo que sean propiedades de un objeto vean. (Administrador, 2014)

Buenas prácticas

Constantes

Solo -1, 0 y 1 se podrán codificar directamente. (Administrador, 2014)

Propiedades

Solo se debe acceder a las propiedades de una clase mediante los métodos get y set. La asignación de variables / propiedades no podrá ser consecutiva. Se debe evitar utilizar el operador asignación para que no se confunda con el operador igualdad en lugares de riesgo. (Administrador, 2014)

Métodos

Es necesario evitar acceder a un método estático desde una instancia de una clase, se debe usar la misma clase. (Administrador, 2014)

Para la implementación de este requerimiento no funcional se ha realizado su respectiva historia técnica, tareas de ingeniería y pruebas de aceptación las cuales se detallan a continuación.

Tabla 41-3: HT-07 Estándar de codificación.

HISTORIA TÉCNICA	
Número: HT-05	Nombre Historia Técnica: Definición del Estándar de Programación.
Usuario: Desarrollador	Sprint Asignado: 01
Fecha Inicio: 08/02/17	Fecha Fin: 09/02/17
Descripción: Como desarrollador necesito definir un estándar de codificación para establecer una norma de cómo escribir el código en el proyecto.	
Pruebas de aceptación: Verificar si el estándar de codificación definido por parte de los miembros de la Fundación y el desarrollador puede ser empleado en el sistema.	

Realizado por: Erika R. 2017

Tabla 42-3: Prueba de aceptación 1 para HT-05

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Historia Técnica: Definición del Estándar de Programación.
Nombre de la Prueba: Verificar si el estándar de codificación definido por parte de los miembros de la Fundación y el desarrollador puede ser empleado en el sistema.	
Responsable: Erika Reina	Fecha: 09/02/17
Descripción: Mediante el estándar de codificación se verificará si se puede emplear dichas normas en el desarrollo del sistema.	
Condiciones de Ejecución: Conocimiento básico de estándares de codificación	
Pasos de Ejecución: Verificar si las normas de codificación desarrolladas se pueden aplicar en el sistema.	
Resultado Esperado: Las normas de codificación están adecuadas para el desarrollo del sistema.	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

Las tareas de ingeniería realizadas en la HT-05 se definen en la siguiente tabla

Tabla 43-3: Tabla de Actividades para HT-05

FECHA	ACTIVIDAD	TIEMPO
08/02/17- 09/02/17	Tarea 1: Definición del estándar de codificación	16 horas

Realizado por: Erika R. 2017

Tabla 44-3: Tarea de Ingeniería 1 para la HT-05

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 01
Nombre Historia Técnica: Definición del Estándar de Programación.	
Nombre de la Tarea: Definición del estándar de codificación.	
Tipo de Tarea: Diseño	Programador Responsable: Erika Reina
Fecha de Inicio: 08/02/17	Fecha Fin: 09/02/17
Descripción: Se definirá un estándar de codificación que permita ser adaptable a futuras mejoras del sistema.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Aprobación del estándar de codificación. 	

Realizado por: Erika R. 2017

Tabla 45-3: Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Definición del Estándar de Programación.
Nombre de la Prueba: Aprobación del estándar de codificación.	
Responsable: Erika Reina	Fecha: 09/02/17
Descripción: Se verificará que el estándar de codificación desarrollado cumpla con las expectativas del Jefe de Sistemas de la Fundación, así como del desarrollador del mismo.	
Condiciones de Ejecución: Conocimiento básico de estándares de codificación.	
Pasos de Ejecución: Presentar al Jefe de Sistemas de la Fundación el estándar desarrollado.	
Resultado Esperado: Aprobación del estándar de codificación.	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

3.1.3.2.3 *Diseño del diagrama de clases*

Se realizó con el fin de especificar la estructura, funcionamiento y diseño del sistema representando las clases que involucra, así como las relaciones existentes entre ellas, el mismo que se constituye como el pilar básico del modelado con UML. Se hizo uso de la herramienta Star UML 5.0.2 para la construcción de las clases que componen el sistema, así se tomó aquellas que son más representativas para el sistema omitiendo aquellas creadas por los servicios de la aplicación, teniendo como resultado 17 clases.

Este diagrama de clases representó a aquellas entidades que componen el sistema “SISTAHU” evidenciando su funcionamiento, en donde se pudo notar que existen composiciones, agregaciones y asociaciones directas entre ciertas tablas.

A continuación, lo podemos visualizar en la **figura 13-3**.

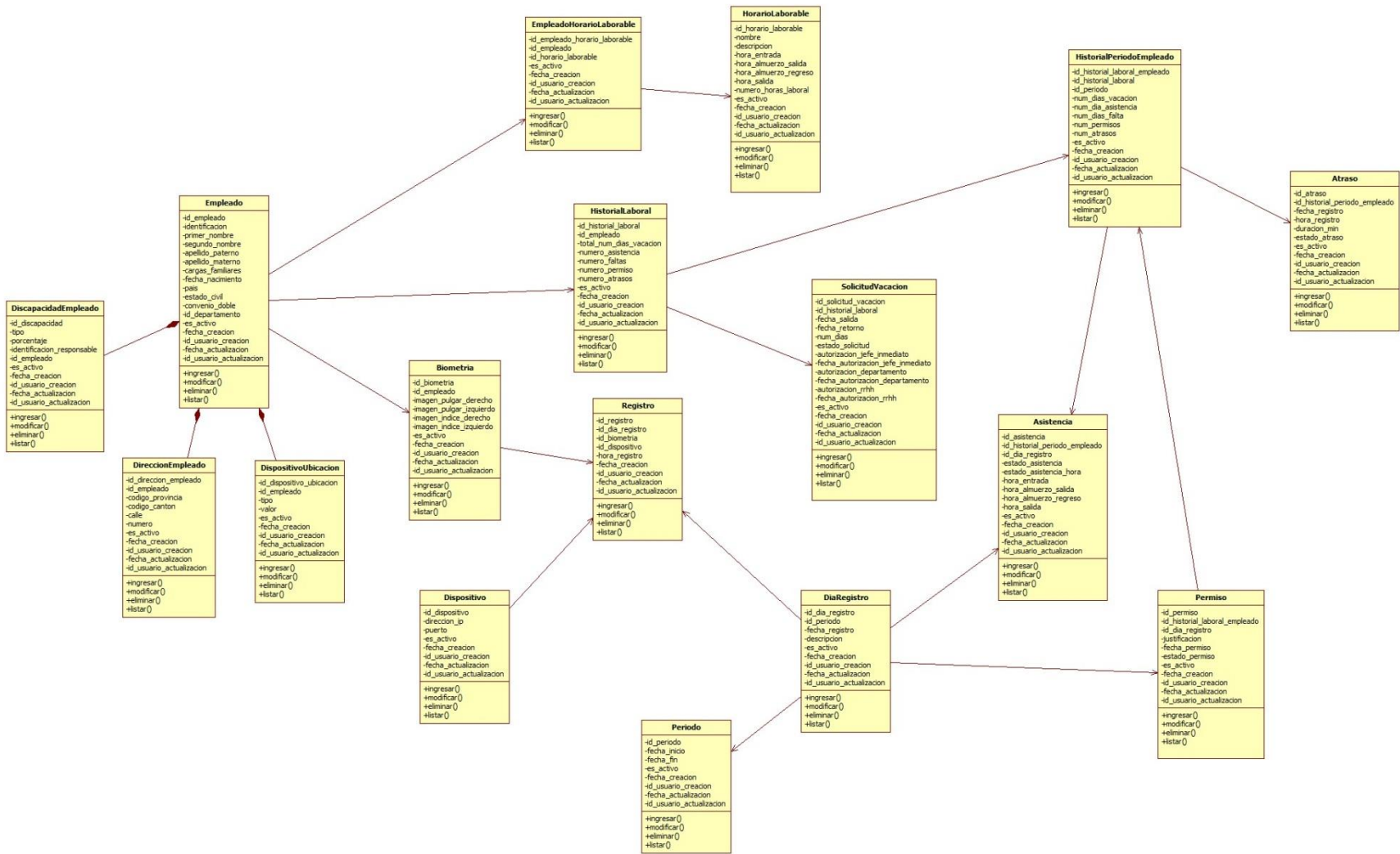


Figura 13-3: Diagrama de Clases “SISTAHU”
 Realizado por: Erika R. 2017

Para la implementación de este requerimiento no funcional se ha realizado su respectiva historia técnica, tareas de ingeniería y pruebas de aceptación las cuales se detallan a continuación.

Tabla 46-3: HT-06 Estándar de codificación.

HISTORIA TÉCNICA	
Número: HT-06	Nombre Historia Técnica: Desarrollo del diagrama de Clases.
Usuario: Desarrollador	Sprint Asignado: 01
Fecha Inicio: 10/02/17	Fecha Fin: 13/02/17
Descripción: Como desarrollador necesito implementar el diagrama de clases que permitirá visualizar la estructura del sistema sus clases, atributos, métodos y las relaciones entre los objetos.	
Pruebas de aceptación: Verificar que el diagrama en UML sea el diagrama aprobado por el jefe de sistemas y el desarrollador.	

Realizado por: Erika R. 2017

Tabla 47-3: Prueba de aceptación 1 para HT-06

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Historia Técnica: Desarrollo del diagrama de Clases.
Nombre de la Prueba: Verificar que el diagrama en UML sea el diagrama aprobado por el jefe de sistemas y el desarrollador.	
Responsable: Erika Reina	Fecha: 13/02/17
Descripción: Se verificará cada una de las clases y sus componentes desarrollados en UML sean los definidos y aprobados previamente	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Conocimiento básico de UML 	
Pasos de Ejecución: Verificar si el diagrama en uml es el modelo aprobado.	
Resultado Esperado: El diagrama UML coincide con el modelo aprobado.	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

Las tareas de ingeniería realizadas en la HT-08 se definen en la siguiente tabla

Tabla 48-3: Tabla de Actividades para HT-06

FECHA	ACTIVIDAD	TIEMPO
10/02/17- 13/02/17	Tarea 1: Creación del diagrama de clases	16 horas

Realizado por: Erika R. 2017

Tabla 49-3: Tarea de Ingeniería 1 para la HT-06

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 01
Nombre Historia Técnica: Desarrollo del diagrama de Clases.	
Nombre de la Tarea: Creación del diagrama de clases	
Tipo de Tarea: Diseño	Programador Responsable: Erika Reina
Fecha de Inicio: 10/02/17	Fecha Fin: 13/02/17
Descripción: Se desarrollará el diagrama de clases UML del sistema.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Aprobación del diagrama de clases. 	

Realizado por: Erika R. 2017

Tabla 50-3: Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Definición del Estándar de Programación.
Nombre de la Prueba: Aprobación del diagrama de clases.	
Responsable: Erika Reina	Fecha: 13/02/17
Descripción: Se verificará que el diagrama de clases desarrollado cumpla con las expectativas del Jefe de Sistemas de la Fundación, así como del desarrollador del mismo.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Tener el modelo Lógico de la Base de datos • Conocimiento básico de uml. 	
Pasos de Ejecución: Presentar al Jefe de Sistemas de la Fundación el diagrama de clases desarrollado.	
Resultado Esperado: Aprobación del diagrama de clases.	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

3.1.3.2.4 Diseño de Interfaces

Definir un estándar para el diseño de las interfaces del sistema permitirá facilitar la creación de las mismas al tener ya establecidos los diferentes colores, fuentes, formas, etc. lo que evitará pérdida de tiempo en la selección de ellos, así como descoordinación entre los diseños de las pantallas. En la **figura 14-3**. podemos visualizar la Página Principal la misma que estará dividida en 3 zonas la cabecera, el cuerpo y el pie de página.

La cabecera se dividirá en 3 secciones: en la parte izquierda irá el logo de la fundación, en el centro estará el nombre del sistema y en la parte derecha se visualizará en login.

El cuerpo tendrá una galería de fotos con una leyenda en la parte posterior de cada foto

El pie de página contendrá 2 columnas; en la columna derecha estará la información de contacto de la Fundación Amiga, el copyright y en la columna izquierda se encontrarán los links con los iconos de facebook y twitter respectivamente.



Figura 14-3: Bocetos de interfaz de usuario (Página Principal)
Realizado por: Erika R. 2017

Una vez dentro del sistema, las pantallas de Gestión de los requisitos funcionales **figura 15-3** estarán divididas en 2 secciones: Cabecera y Cuerpo.

La cabecera contendrá el Logo de la Fundación un MenuBar con las opciones respectivas de tipo de usuario, cada opción tendrá un ícono representativo.

El cuerpo tendrá la en primer plano una grilla la cual contendrá los datos de la opción seleccionada junto a cada registro estarán los iconos de editar y eliminar, en la parte superior de los registros habrá filtros de búsqueda por columnas. Existirá un filtro General el cual se encontrará en la parte superior derecha y finalmente en la parte superior centro se encuentra el botón para agregar un nuevo registro.



Figura 15-3: Pantalla Funcional General
Realizado por: Erika R. 2017

Cada especificación de colores, tipo de letras, ubicación de íconos, etc. fue estructurado de acuerdo a los parámetros establecidos por el presidente de la Fundación Amiga (StakeHolder) en conjunto con el desarrollador del sistema.

Para la implementación de este requerimiento no funcional se ha realizado su respectiva historia técnica, tareas de ingeniería y pruebas de aceptación las cuales están desarrolladas a continuación.

Tabla 51-3: HT-07 Bocetos de la interfaz de usuario

HISTORIA TÉCNICA	
Número: HT-07	Nombre Historia Técnica: Diseño de Interfaces.
Usuario: Desarrollador	Sprint Asignado: 01
Fecha Inicio: 16/02/17	Fecha Fin: 23/02/17
Descripción: Como desarrollador necesito establecer bocetos de la interfaz de usuario creando un estándar para cada interfaz requerida.	

Pruebas de aceptación:

- Verificar que las pantallas del sistema mantengan la imagen corporativa de la Fundación Amiga.
- Diseñar los formularios de ingreso, modificación y eliminación.

Realizado por: Erika R. 2017

Tabla 52-3 Prueba de aceptación 1 de la HT-07

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Historia Técnica: Establecer bocetos de la interfaz de usuario.
Nombre de la Prueba: Verificar que las pantallas del sistema mantengan la imagen corporativa de la Fundación Amiga.	
Responsable: Erika Reina	Fecha: 23/02/17
Descripción: Los bocetos o diseños presentados mantengan la imagen corporativa de la Fundación Amiga.	
Condiciones de Ejecución <ul style="list-style-type: none"> • Contar con Imagen corporativa de la Fundación Amiga. 	
Pasos de Ejecución: <ul style="list-style-type: none"> • Combinar los colores de la Fundación Amiga. • Ubicar las posiciones de los elementos que va tener el sistema. 	
Resultado Esperado: Los diseños o bocetos se relacionan con la imagen corporativa de la Fundación Amiga.	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

Las tareas de ingeniería realizadas en la HU-07 se definen en la siguiente tabla

Tabla 53-3: Tabla de Actividades para HU-07

FECHA	ACTIVIDAD	TIEMPO
14/02/17 - 14/02/17	Tarea 1: Diseño del tema a utilizar en la interfaz de usuario.	8 horas
15/11/16 - 15/02/17	Tarea 2: Diseño de la interfaz de usuario.	8 horas

Realizado por: Erika R. 2017

Tabla 54-3: Tarea de Ingeniería 1 para la HT-09

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 01
Nombre de Historia Técnica: Diseño de Interfaces.	
Nombre de la Tarea: Diseño del tema a utilizar en la interfaz de usuario.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina
Fecha de Inicio: 14/02/17	Fecha Fin: 14/02/17
Descripción: Se realizará el diseño del tema que se utilizará en la interfaz del sistema, basado en las especificaciones de los usuarios.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Aprobación del tema desarrollado. 	

Realizado por: Erika R. 2017

Tabla 55-3: Prueba de aceptación 1 para tara de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Diseño de Interfaces.
Nombre de la Prueba: Aprobación del tema desarrollado.	
Responsable: Erika Reina	Fecha: 15/02/17
Descripción: Diseñar un tema acorde a las especificaciones de los usuarios.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Investigación de temas existentes. • Consultar con el Gerente sobre temas propuestos. 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> • Diseño del tema. • Presentar el tema a los usuarios. 	
Resultado Esperado: Aprobación del tema desarrollado.	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Erika R. 2017

Tabla 56-3: Tarea de Ingeniería 2 para la HT-07

TAREAS DE INGENIERÍA	
SPRINT: 01	Número de Tarea: 02
Nombre de Historia Técnica: Diseño de interfaces.	
Nombre de la Tarea: Diseño de la interfaz de usuario.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina
Fecha de Inicio: 15/02/17	Fecha Fin: 15/02/17
Descripción: Diseñar la interfaz de usuario con las características especificadas, indicándole sugerencias para una buena interfaz.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Aprobación del diseño de interfaz del sistema. 	

Realizado por: Erika R. 2017

Tabla 57-3: Prueba de aceptación 1 para tarea de ingeniería 2.

PRUEBAS DE ACEPTACIÓN	
Código: 2.1	Nombre Tarea de Ingeniería: Diseño de Interfaces.
Nombre de la Prueba: Aprobación del diseño de interfaz del sistema.	
Responsable: Erika Reina	Fecha: 15/02/17
Descripción: Verificar que el diseño de la interfaz de usuario sea amigable y cumpla con las expectativas del usuario.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Haber revisado bosquejos de diseños de Interfaz de Usuario anteriores. • Haber tomado en cuenta los requerimientos de diseño del usuario 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> • Diseño de la interfaz de usuario. • Presentar la interfaz a los usuarios. 	
Resultado Esperado: Aprobación del diseño de interfaz.	
Evaluación de la Prueba:	
<ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

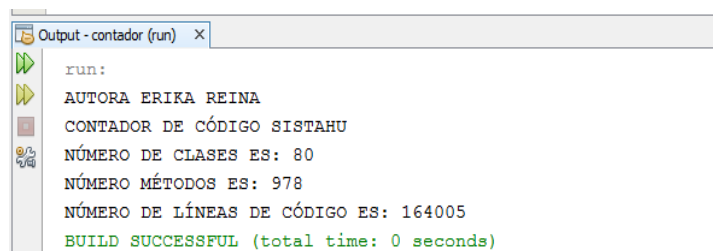
Codificación

Con el objetivo de programar el sistema “SISTAHU”, en el lenguaje de programación específico, se desarrolló en orden de prioridad las historias de usuario, tal como indica el Plan de Entrega, las mismas que inician en el Sprint 2 hasta el Sprint 10.

Cada historia de usuario desarrollada desempeña un requisito funcionalidad específico del sistema, estos plasman en teoría la codificación de los requerimientos establecidos por el usuario. Para el desarrollo del sistema se ha utilizado el IDE Netbeans 8.1, JDK 1.8 bajo el lenguaje de programación java, framework jsf y el servidor GlassFish 4.1.1.

En el **Anexo 4** se presentan las Historias de Usuario del sistema “SISTAHU” cada una con sus respectivas tareas de Ingeniería y estas a su vez cuentan con sus pruebas de aceptación respectivas.

Una vez realizado la implementación del sistema se realizó un conteo de las clases, métodos y las líneas de código del que está constituido el sistema, a continuación, en la **figura 16-3**. se presenta el resultado de la ejecución del programa contador.



```
run:
AUTORA ERIKA REINA
CONTADOR DE CÓDIGO SISTAHU
NÚMERO DE CLASES ES: 80
NÚMERO MÉTODOS ES: 978
NÚMERO DE LÍNEAS DE CÓDIGO ES: 164005
BUILD SUCCESSFUL (total time: 0 seconds)
```

Figura 16-3: Contador de Código SISTAHU
Realizado por: Erika R. 2017

Con la ayuda del sistema informático contador de código SISTAHU se presentó un total de 80 clases; en donde se incluyen las 17 establecidas en el diagrama de clases del sistema, además de las clases generadas por hibernate en conjunto con primefaces. A su vez se determinó 978 métodos, así como 164005 líneas de código.

3.1.4 Fase de Cierre

3.1.4.1 Evaluación de la gestión del proyecto

La gestión del proyecto permitió planificar, dirigir y controlar el desarrollo del mismo, a bajo costo y cumpliendo con el tiempo establecido para su culminación. Una vez que el sistema finalizó su desarrollo se pudo realizar un análisis en cuanto a la productividad del mismo para ello se empleó un programa contador de código.

Con los datos obtenidos del contador de código SISTAHU se pudo determinar la velocidad del proyecto a través del cálculo de la productividad. Para ello se realizó una división entre las líneas de código escritas y las horas de programación efectuadas por el desarrollador del sistema, lo que da como resultado la productividad del mismo.

Determinar la productividad del developer team permitió que se puedan tomar decisiones con respecto a su nivel de desarrollo y así constatar si se produjo una mejora o no en el proceso y como tendrá beneficios o repercusiones en el proyecto.

Productividad= líneas de código / (horas de programación)

Productividad= 164005/(720)

Productividad= 227.78 lc/h.

Para evidenciar esta gestión; es decir el nivel de cumplimiento correspondiente a lo planificado se usó el diagrama Burn Down Chart que se visualiza en la **Gráfico 8-3**.

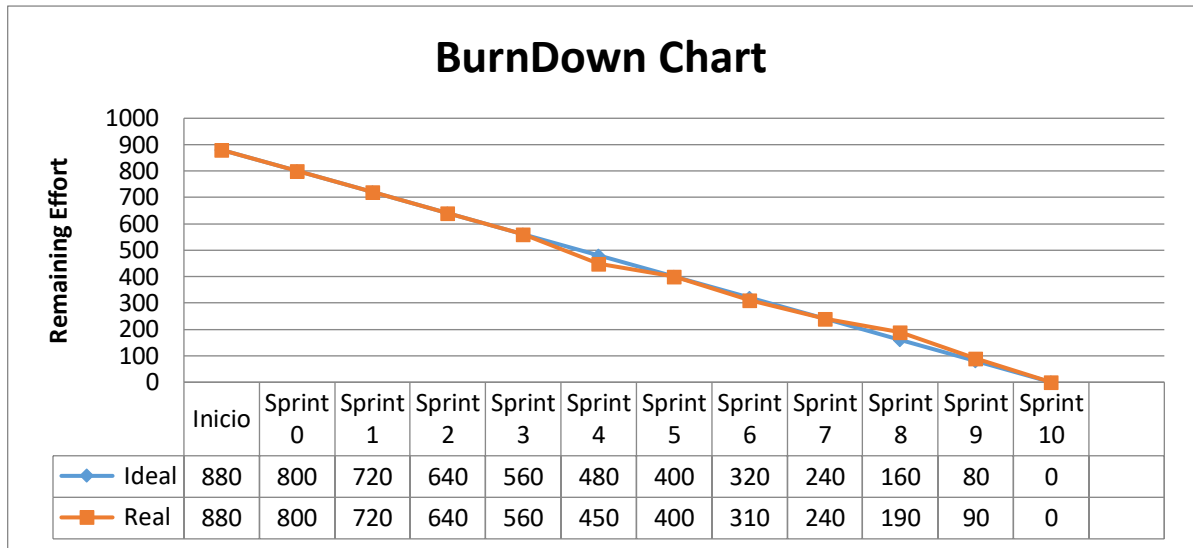


Gráfico 8-3: Contador de Código "SISTAHU"

Realizado por: Erika R. 2017

En el **Gráfico 8-3**, se puede visualizar que iniciamos el proyecto con un total de 880 puntos y a medida que se concluye un sprint la cantidad de puntos se reduce; es decir se disminuye el valor del sprint al total de puntos restantes.

Existe, en el sprint 4 y 6, una velocidad del proyecto de 30 y 10 puntos estimados respectivamente con una mejora significativa debido a que se contaba con conocimientos altos de las herramientas de desarrollo, además hubo experiencia en programación similar a los sprints anteriores, por lo que se pudo avanzar mucho más rápido.

Sin embargo, en los sprints 8 y 9 existió un retraso de 40 puntos estimados debido a que se desconocían los procesos que se implicaban en los sprints respectivos, por lo que se pudo compensar este tiempo con el avance significativo que se tuvo en los sprints mencionados anteriormente, por lo que el proyecto culminó en la fecha de entrega establecida en la planificación.

3.1.4.2 Manual de Usuario

Este manual permitirá a cada uno de los usuarios que manejarán el sistema tener una guía clara de cómo deberá usar el mismo, así como sus funcionalidades y características.

Se ha realizado un documento en Microsoft Word 2016 que consta de 40 hojas de contenido, que detalla los pasos con su respectiva imagen de cada una de las opciones del sistema, para que los usuarios los cuales son: el administrador y el encargado de recursos humanos puedan manejar el sistema “SISTAHU” de tal forma que no exista inconvenientes en su uso. El manual de usuario se encuentra adjunto al presente trabajo de titulación.

3.2 Determinación de atributos no funcionales

3.2.1 Requisitos no funcionales

3.2.1.1 Seguridad

- El sistema brinda seguridad ante el acceso de entes externos mediante un esquema de autenticación con nombre de usuario y una clave.

3.2.1.2 Disponibilidad

- El sistema estará disponible las 24 horas del día los 7 días de la semana los 365 días del año, debido a que es un sistema web distribuido; es decir la aplicación cuenta con dos componentes que se ejecutan en entornos separados.

3.2.1.3 Eficiencia

- Determinada en el *ítem 3.3*

3.2.1.4 Usabilidad

- El sistema es fácil de usar por su alto nivel de intuitividad, esto se determinó mediante el manejo del sistema por el presidente de la fundación y al encargado de recursos humanos,

administradores del manejo del mismo, en el cual se establecieron 3 parámetros de usabilidad satisfactoriamente presentados en la **tabla 58-3**.

Tabla 58-3: Resultados Parámetros de Usabilidad

Parámetros de usabilidad	Usuario 1	Usuario 2
El sistema es entendible para el uso de determinadas tareas.	SI	SI
El sistema es fácil de aprender su funcionamiento.	SI	SI
El sistema es fácil para su manejo y control.	SI	SI

Realizado por: Erika Reina. 2017

3.3 Determinación de la Eficiencia Interna y externa del sistema SISTAHU

Basado en los formatos establecidos y especificados en la norma ISO/IEC 9126-2 Y 9126-3 se tomaron los datos requeridos para cada uno de los requerimientos a medir los cuales se encuentran descritos en el **ítem 2.3.4**.

Dando como resultado lo siguiente:

Característica: Eficiencia interna

Subcaracterística: Utilización de recursos.

Métrica: Utilización I/O densidad de mensaje

Tabla 59-3: Resultados Métrica Interna Utilización I/O Densidad de mensaje

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala de métrica	Tipo de Medida	Entradas para medición	Usuarios Seleccionados
Utilización I/O Densidad de mensaje	¿Cuál es la densidad de mensajes relacionado con la utilización de I/O en las líneas de código responsables hacienda llamadas del Sistema?	Contar el número de errores que pertenecen a fallas de I/O, advertencia y comparar al número estimado de líneas de código responsable en llamadas al sistema	<ol style="list-style-type: none"> 1. A=4 B=580 X=4/580 X=0.0069 2. A=1 B=510 X=1/510 X=0.0020 3. A=2 B=556 X=2/556 X=0.0036 4. A=2 B=629 X=2/629 X=0.0032 5. A=1 B=191 X=1/191 X=0.0052 	0.0069	Absoluto	X=contable/contable A=contable B=contable	Código Fuente	Diseñadores

Realizado por: Erika Reina, 2017

Fuente: (Vivanco Villamar, 2011: p.111)

Para la toma de datos se realizó el siguiente proceso:

Los errores controlados se evaluaron en los controladores que intervinieron en las tareas en la capa de presentación de la aplicación:

Controlador

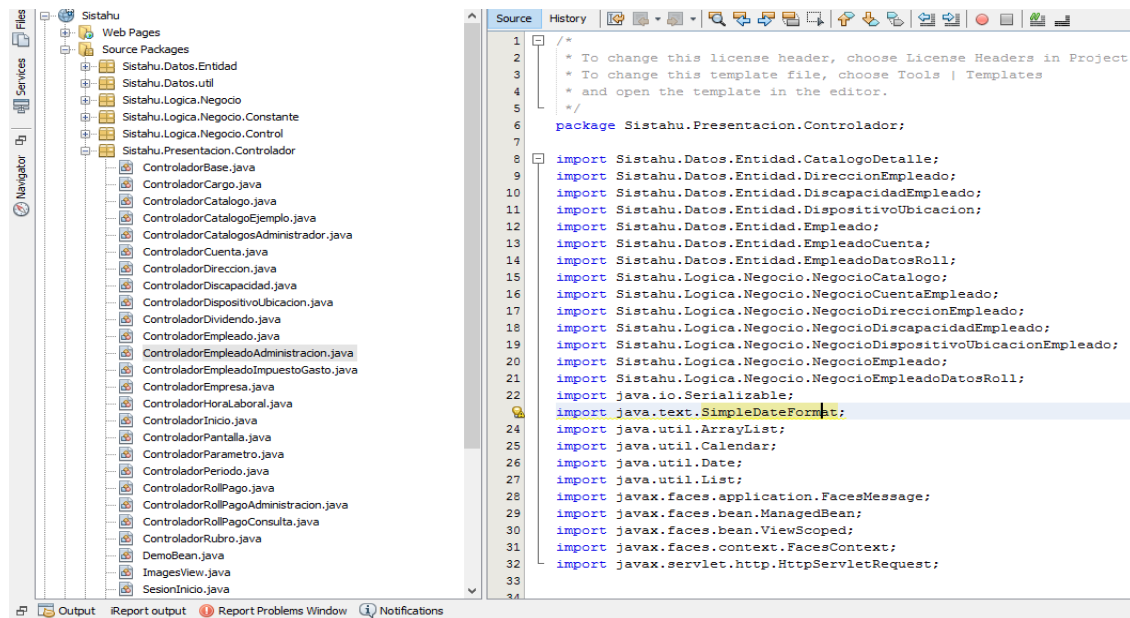


Figura 17-3: Controlador SISTAHU

Realizado por: Erika R. 2017

Evaluación prevista o error controlado

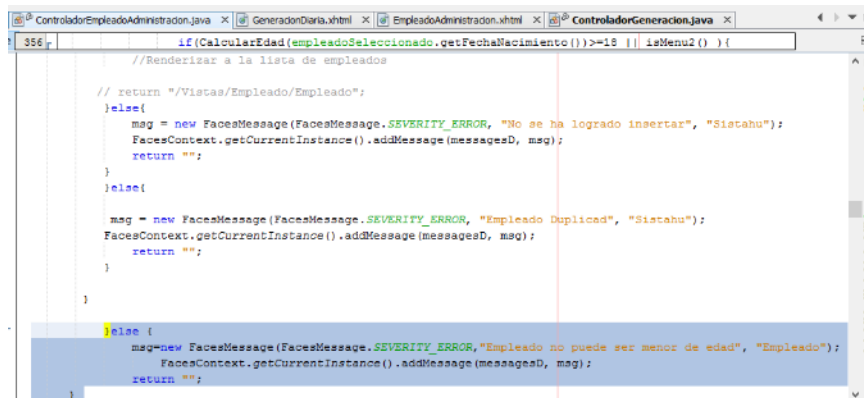


Figura 18-3: Errores controlados

Realizado por: Erika R. 2017

1) Registrar datos del empleado

Total, errores controlados: 4

1.1) Error de empleado menor de edad

```
if (validacionEmpleado.getEmpleado().getEdad() < 18) { return true; }
negocioEmpleadoDatosRoll.guardar(datosRolEmpleadoSeleccionado);
}

//Renderizar a la lista de empleados

// return "/Vistas/Empleado/Empleado":
}else{
    System.err.println("No es posible insertar direcciones");
}
}else{
    msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Empleado Duplicad", "
FacesContext.getCurrentInstance().addMessage(messagesD, msg);
    return "";
}
}

}else {
    msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Empleado no puede ser
FacesContext.getCurrentInstance().addMessage(messagesD, msg);
    return "";
}
```

Figura 19-3: Error de empleado menor de edad
Realizado por: Erika R. 2017

1.2) Error de duplicidad empleado

```
if(empleadoDuplicado==null){
    dispositivoUbicacion.setEmpleado(empleadoSeleccionado);
    negocioDispositivoUbicacionEmpleado.guardar(dispositivoUbicacionEmpleado);
}

if(datosRolEmpleadoSeleccionado!=null){
    datosRolEmpleadoSeleccionado.setEmpleado(empleadoSeleccionado);
    negocioEmpleadoDatosRoll negocioEmpleadoDatosRoll= new negocioEmpleadoDatosRoll();
    negocioEmpleadoDatosRoll.guardar(datosRolEmpleadoSeleccionado);
}

//Renderizar a la lista de empleados

// return "/Vistas/Empleado/Empleado":
}else{
    System.err.println("No es posible insertar direcciones");
}
}else{
    msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Empleado Duplicad", "
FacesContext.getCurrentInstance().addMessage(messagesD, msg);
    return "";
}
}

}else {
    msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Empleado no puede ser
FacesContext.getCurrentInstance().addMessage(messagesD, msg);
}
```

Figura 20-3: Error de duplicidad empleado
Realizado por: Erika R. 2017

1.3) Error de inserción de empleado

```
if (empleadoSeleccionado.getIdEmpleado() > 0) {
    dispositivoUbicacion.setEmpleado(empleadoSeleccionado);
    negocioDispositivoUbicacionEmpleado.guardar(dispositivoUbicacionEmpleado);
}

if (datosRolEmpleadoSeleccionado != null) {
    datosRolEmpleadoSeleccionado.setEmpleado(empleadoSeleccionado);
    negocioEmpleadoDatosRoll negocioEmpleadoDatosRoll = new negocioEmpleadoDatosRoll();
    negocioEmpleadoDatosRoll.guardar(datosRolEmpleadoSeleccionado);
}

//Renderizar a la lista de empleados.
// return "/Vistas/Empleado/Empleado";
} else {
    msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "No se ha logrado insertar el empleado.");
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
    return "";
}
} else {
    msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Empleado Duplicado", "Empleado Duplicado");
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
    return "";
}
}

} else {
```

Figura 21-3: Error de inserción de empleado

Realizado por: Erika R. 2017

1.4) Error al no contar con datos para guardar

```
if (this.empleadoSeleccionado != null) {
    } else {
        msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Empleado Duplicado", "Empleado Duplicado");
        FacesContext.getCurrentInstance().addMessage(messagesD, msg);
        return "";
    }

    } else {
        msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Empleado no puede ser guardado", "Empleado no puede ser guardado");
        FacesContext.getCurrentInstance().addMessage(messagesD, msg);
        return "";
    }

} else {
    msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Sin datos", "Empleado");
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
    return "";
}

} catch (Exception e) {
    System.err.println("Error al crear el empleado: "+e.getMessage());
}

return "Empleado.xhtml?faces-redirect=true";
}
```

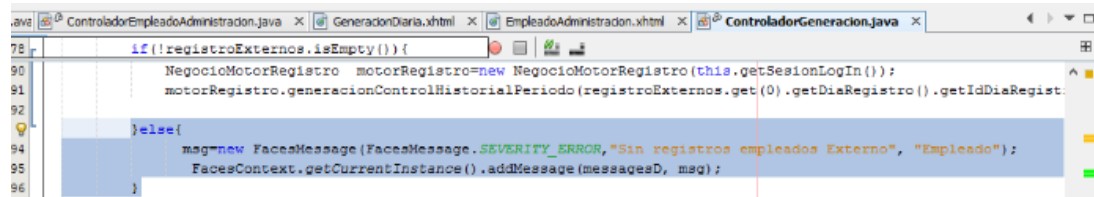
Figura 22-3: Error al no contar con datos para guardar

Realizado por: Erika R. 2017

2) Generar los días de vacaciones a partir del registro diario

Total, Errores controlados: 1

2.1) Error al no contar con registros en la fuente externa para ser cargada en la base de datos local



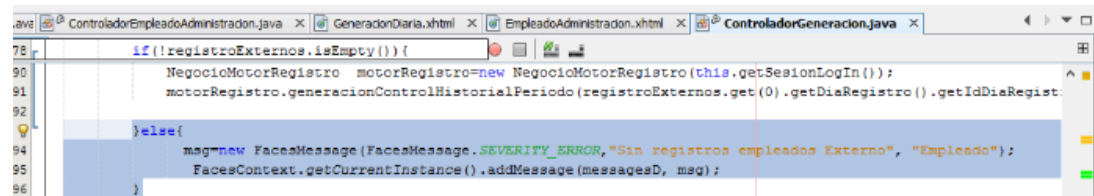
```
78 | if(!registroExternos.isEmpty()){
90 |     NegocioMotorRegistro motorRegistro=new NegocioMotorRegistro(this.getSesionLogIn());
91 |     motorRegistro.generacionControlHistorialPeriodo(registroExternos.get(0).getDiaRegistro().getIdDiaRegist:
92 |
93 | }else{
94 |     msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin registros empleados Externo", "Empleado");
95 |     FacesContext.getCurrentInstance().addMessage(messagesD, msg);
96 | }
```

Figura 23-3: Error al no contar con registros para vacaciones
Realizado por: Erika R. 2017

3) Generar atrasos a partir del registro diario

Total, Errores no controlados: 2

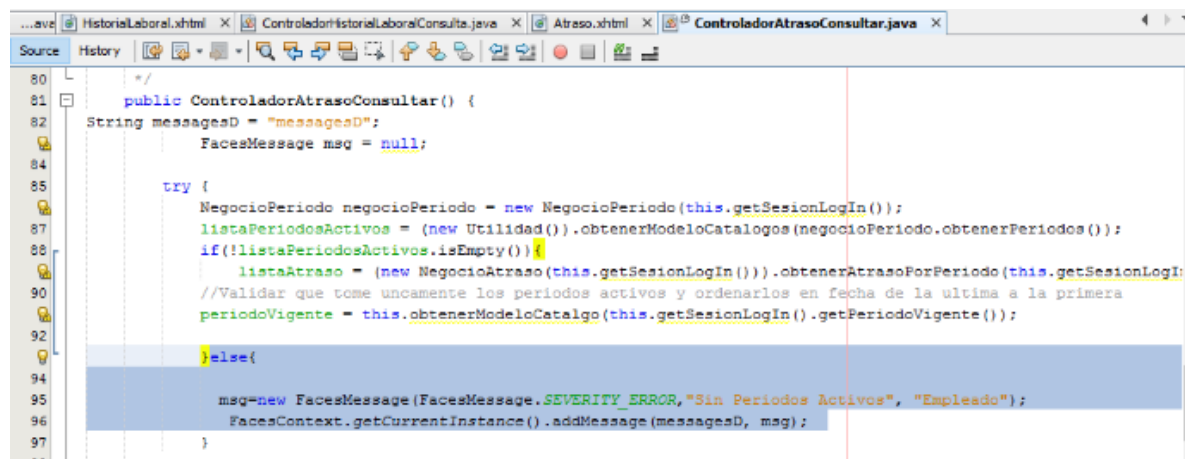
3.1) Error al no contar con registros en la fuente externa para ser cargada en la base de datos local



```
78 | if(!registroExternos.isEmpty()){
90 |     NegocioMotorRegistro motorRegistro=new NegocioMotorRegistro(this.getSesionLogIn());
91 |     motorRegistro.generacionControlHistorialPeriodo(registroExternos.get(0).getDiaRegistro().getIdDiaRegist:
92 |
93 | }else{
94 |     msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin registros empleados Externo", "Empleado");
95 |     FacesContext.getCurrentInstance().addMessage(messagesD, msg);
96 | }
```

Figura 24-3: Error al no contar con registros para atrasos
Realizado por: Erika R. 2017

3.2) Error al no existe periodos activos



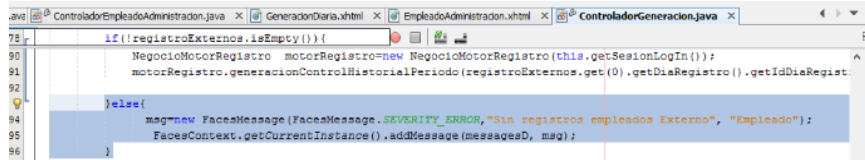
```
80 |
81 | public ControladorAtrasoConsultar() {
82 |     String messagesD = "messagesD";
83 |     FacesMessage msg = null;
84 |
85 |     try {
86 |         NegocioPeriodo negocioPeriodo = new NegocioPeriodo(this.getSesionLogIn());
87 |         listaPeriodosActivos = (new Utilidad()).obtenerModeloCatalogos(negocioPeriodo.obtenerPeriodos());
88 |         if(!listaPeriodosActivos.isEmpty()){
89 |             listaAtraso = (new NegocioAtraso(this.getSesionLogIn())).obtenerAtrasoPorPeriodo(this.getSesionLogIn:
90 |             //Validar que tome uncamente los periodos activos y ordenarlos en fecha de la ultima a la primera
91 |             periodoVigente = this.obtenerModeloCatalogo(this.getSesionLogIn().getPeriodoVigente());
92 |         }else{
93 |
94 |
95 |             msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin Periodos Activos", "Empleado");
96 |             FacesContext.getCurrentInstance().addMessage(messagesD, msg);
97 |         }
98 |     }
```

Figura 25-3: Error al no existe periodos activos
Realizado por: Erika R. 2017

4) Generar asistencia a partir del registro diario

Total, errores no controlados: 2

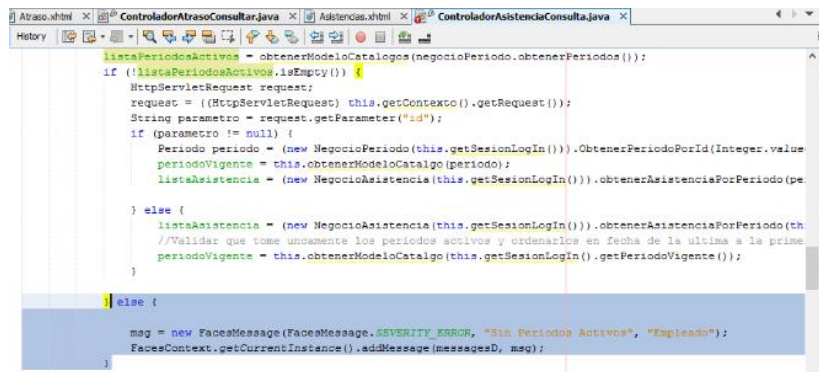
4.1) Error al no contar con registros en la fuente externa para ser cargada en la base de datos local



```
78 if(!registroExternos.isEmpty()){
90     NegocioMotorRegistro motorRegistro=new NegocioMotorRegistro(this.getSessionLogIn());
91     motorRegistro.generacionControlHistorialPeriodo(registroExternos.get(0).getDiaRegistro().getDiaRegist
92 }
93 }else{
94     msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin registros empleados Externo", "Empleado");
95     FacesContext.getCurrentInstance().addMessage(messagesD, msg);
96 }
```

Figura 26-3: Error al no contar con registros para asistencia
Realizado por: Erika R. 2017

4.2) Error al no existir periodos activos



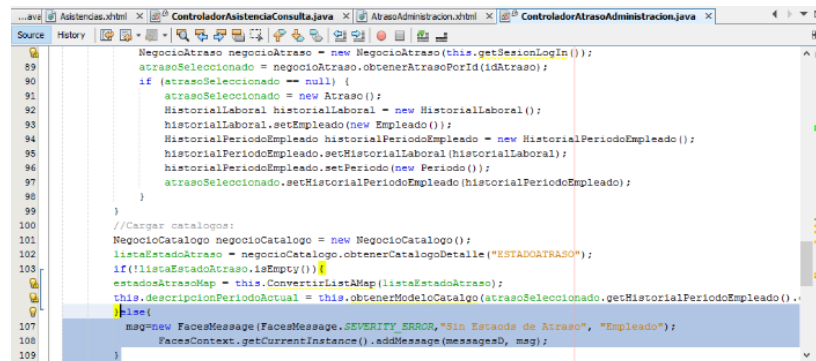
```
listaPeriodosActivos = obtenerModeloCatalogos(negocioPeriodo.obtenerPeriodos());
if (!listaPeriodosActivos.isEmpty()) {
    HttpServletRequest request;
    request = (HttpServletRequest) this.getContexto().getRequest();
    String parametro = request.getParameter("id");
    if (parametro != null) {
        Periodo periodo = (new NegocioPeriodo(this.getSessionLogIn())).ObtenerPeriodoPorId(Integer.valueOf
        periodoVigente = this.obtenerModeloCatalogo(periodo);
        listaAsistencia = (new NegocioAsistencia(this.getSessionLogIn())).obtenerAsistenciaPorPeriodo(pe
    } else {
        listaAsistencia = (new NegocioAsistencia(this.getSessionLogIn())).obtenerAsistenciaPorPeriodo(th
        //Validar que tome unicamente los periodos activos y ordenarlos en fecha de la ultima a la prime
        periodoVigente = this.obtenerModeloCatalogo(this.getSessionLogIn().getPeriodoVigente());
    }
} else {
    msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Sin Periodos Activos", "Empleado");
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
}
```

Figura 27-3: Error al no existir periodos activos
Realizado por: Erika R. 2017

5) Registrar permisos por tipo

Total, errores controlados: 1

5.1) Error al no contar con los estados del atraso



```
90 NegocioAtraso negocioAtraso = new NegocioAtraso(this.getSessionLogIn());
91 atrasoSeleccionado = negocioAtraso.obtenerAtrasoPorId(idAtraso);
92 if (atrasoSeleccionado == null) {
93     atrasoSeleccionado = new Atraso();
94     HistorialLaboral historialLaboral = new HistorialLaboral();
95     historialLaboral.setEmpleado(new Empleado());
96     HistorialPeriodoEmpleado historialPeriodoEmpleado = new HistorialPeriodoEmpleado();
97     historialPeriodoEmpleado.setHistorialLaboral(historialLaboral);
98     historialPeriodoEmpleado.setPeriodo(new Periodo());
99     atrasoSeleccionado.setHistorialPeriodoEmpleado(historialPeriodoEmpleado);
100 }
101 //Cargar catalogos:
102 NegocioCatalogo negocioCatalogo = new NegocioCatalogo();
103 listaEstadoAtraso = negocioCatalogo.obtenerCatalogoDetalle("ESTADOATRASO");
104 if (!listaEstadoAtraso.isEmpty()) {
105     estadosAtrasoMap = this.ConvertirListaAMap(listaEstadoAtraso);
106     this.descripcionPeriodoActual = this.obtenerModeloCatalogo(atrasoSeleccionado.getHistorialPeriodoEmpleado().
107 } else {
108     msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin Estado de Atraso", "Empleado");
109     FacesContext.getCurrentInstance().addMessage(messagesD, msg);
110 }
```

Figura 28-3: Error al no contar con los estados del atraso
Realizado por: Erika R. 2017

Característica: Eficiencia interna

Subcaracterística: Utilización de recursos.

Métrica: Utilización de memoria densidad de mensaje

Tabla 60-3: Resultados métrica interna utilización de memoria densidad de mensaje

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala de métrica	Tipo de Medida	Entradas para medición	Usuarios Seleccionados
Utilización de Memoria Densidad de mensaje	¿Cuál es la densidad de mensajes relacionado con la utilización de memoria en las líneas de código responsables haciendo llamadas del Sistema?	Contar el número de mensajes de errores que pertenecen al fallo de memoria y advertencia y comparar el número estimado de líneas de código responsable en llamadas del sistema	<ol style="list-style-type: none"> 1. A=5 B=580 X=5/580 X=0.0086 2. A=3 B=510 X=3/510 X=0.0059 3. A=3 B=556 X=3/556 X=0.0054 4. A=3 B=629 X=3/629 X=0.0048 5. A=4 B=191 X=4/191 X=0.0209 	0.0209	La proporción	X=contable/contable A=contable B=contable	Código Fuente	Diseñadores

Realizado por: Erika Reina. 2017

Fuente: (Vivanco Villamar, 2011: p.112)

Estos errores con capturados mediante la sentencia TRY-CATCH y se evalúan uno por cada método en los controladores involucrados

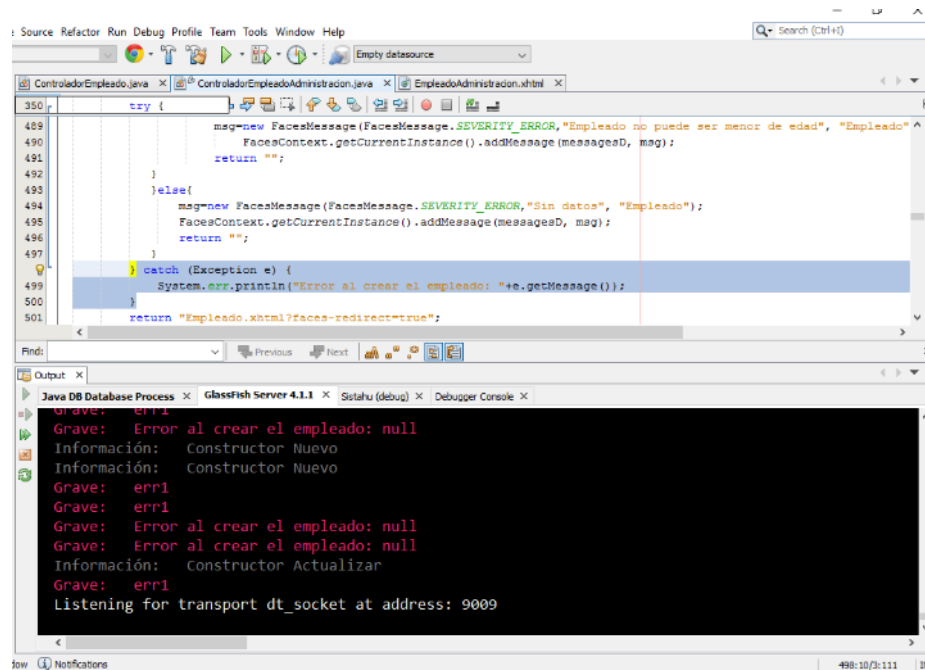


Figura 29-3: Errores con capturados mediante la sentencia TRY-CATCH
Realizado por: Erika R. 2017

1) Registrar datos del empleado

Total, Errores no controlados: 5

1.1) Error al crear o registrar el empleado, null

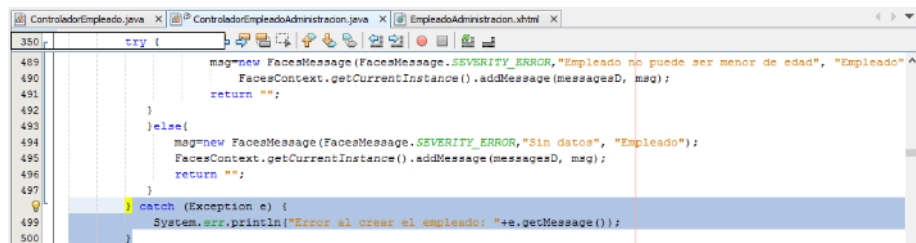


Figura 30-3: Error al crear o registrar el empleado, null
Realizado por: Erika R. 2017

1.2) Error al crear dirección de empleado

```
666 try {
676     }
677 }
678 } else if (discapacidadEmpleadoSeleccionado != null) {
679     this.listaDiscapacidadEmpleado.remove(discapacidadEmpleadoSeleccionado);
680     this.listaDiscapacidadEmpleado.add(discapacidadEmpleadoSeleccionado);
681     //empleadoSeleccionado.getDireccionesEmpleado().add(direccionEmpleadoSeleccionado);
682 }
683 } catch (Exception e) {
684     System.err.println("Error al crear discapacidad del empleado:" + e.getMessage());
685 }
686 }
```

Figura 31-3: Error al crear dirección de empleado
Realizado por: Erika R. 2017

1.3)Error al crear discapacidad del empleado

```
43 }
44 } catch (Exception e) {
45     System.err.println("Error al crear direccion del empleado:" + e.getMessage());
46 }
47 }
```

Figura 32-3: Error al crear discapacidad del empleado
Realizado por: Erika R. 2017

1.4) Error al crear cuenta empleado

```
705 try {
712     this.listaCuentaEmpleado.add(cuentaEmpleadoSeleccionado);
713     NegocioCuentaEmpleado negocioCuentaEmpleado = new NegocioCuentaEmpleado(this.sesionLogIn);
714     negocioCuentaEmpleado.guardar(cuentaEmpleadoSeleccionado);
715 }
716 }
717 } else if (cuentaEmpleadoSeleccionado != null) {
718     this.listaCuentaEmpleado.remove(cuentaEmpleadoSeleccionado);
719     this.listaCuentaEmpleado.add(cuentaEmpleadoSeleccionado);
720     //empleadoSeleccionado.getDireccionesEmpleado().add(direccionEmpleadoSeleccionado);
721 }
722 } catch (Exception e) {
723     System.err.println("Error al crear cuenta del empleado:" + e.getMessage());
724 }
```

Figura 33-3: Error al crear cuenta empleado
Realizado por: Erika R. 2017

1.5) Error al grabar dispositivos ubicación empleado

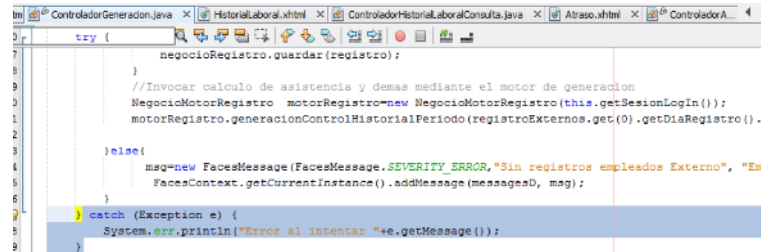
```
744 try {
751     this.listaDispositivoUbicacionEmpleado.add(dispositivoUbicacionEmpleadoSeleccionado);
752     NegocioDispositivoUbicacionEmpleado negocioDispositivoUbicacionEmpleado = new NegocioDispositivoUbicacionEmpleado(this.sesionLogIn);
753     negocioDispositivoUbicacionEmpleado.guardar(dispositivoUbicacionEmpleadoSeleccionado);
754 }
755 }
756 } else if (discapacidadEmpleadoSeleccionado != null) {
757     this.listaDispositivoUbicacionEmpleado.remove(dispositivoUbicacionEmpleadoSeleccionado);
758     this.listaDispositivoUbicacionEmpleado.add(dispositivoUbicacionEmpleadoSeleccionado);
759 }
760 } catch (Exception e) {
761     System.err.println("Error al crear dispositivo ubicación del empleado:" + e.getMessage());
762 }
763 }
```

Figura 34-3: Error al grabar dispositivos ubicación empleado
Realizado por: Erika R. 2017

2) Generar los días de vacaciones a partir del registro diario

Total, errores no controlados: 3

2.1) Error al intentar insertar registros externos a locales



```
try {
    negocioRegistro.guardar(registro);
    //Invocar calculo de asistencia y demas mediante el motor de generacion
    NegocioMotorRegistro motorRegistro=new NegocioMotorRegistro(this.getSessionLogIn());
    motorRegistro.generacionControlHistorialPeriodo(registroExternos.get(0).getDiaRegistro());
}
else{
    msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin registros empleados Externo", "Emp
    FacesContext.getCurrentInstance().addMessage(messageD, msg);
}
} catch (Exception e) {
    System.err.println("Error al intentar "+e.getMessage());
}
```

Figura 35-3: Error al intentar insertar registros externos a locales
Realizado por: Erika R. 2017

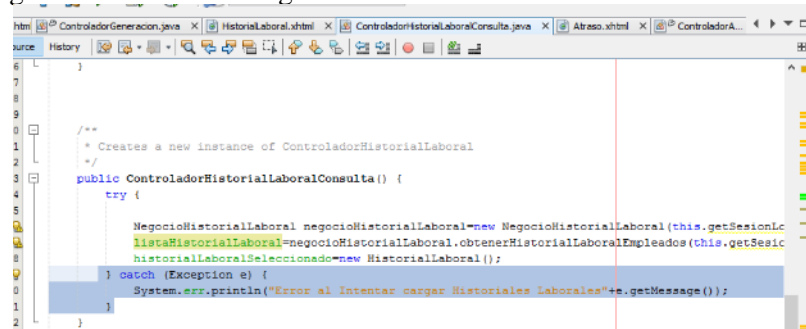
2.2) Error al generar control biométrico vacaciones



```
@PostConstruct
public void PostConstructGenerator() {
    try {
        this.GeneraControlBiotriamicoDiarioEmpleados(this.getSessionLogIn().getFechaSistema().getTime());
    } catch (Exception e) {
        System.err.println("Error al generar control biometrico diario"+e.getMessage());
    }
}
```

Figura 36-3: Error al generar control biométrico vacaciones
Realizado por: Erika R. 2017

2.3) Error al cargar días de vacaciones generadas



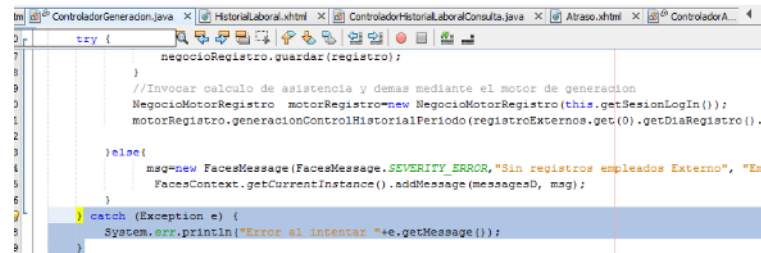
```
/**
 * Creates a new instance of ControladorHistorialLaboral
 */
public ControladorHistorialLaboralConsulta() {
    try {
        NegocioHistorialLaboral negocioHistorialLaboral=new NegocioHistorialLaboral(this.getSessionLogIn());
        listaHistorialLaboral=negocioHistorialLaboral.obtenerHistorialLaboralEmpleados(this.getSessionLogIn().getFechaSistema().getTime());
        historialLaboralSeleccionado=new HistorialLaboral();
    } catch (Exception e) {
        System.err.println("Error al intentar cargar Historiales Laborales"+e.getMessage());
    }
}
```

Figura 37-3: Error al cargar días de vacaciones generadas
Realizado por: Erika R. 2017

3) Generar atrasos a partir del registro diario

Total, errores no controlados: 3

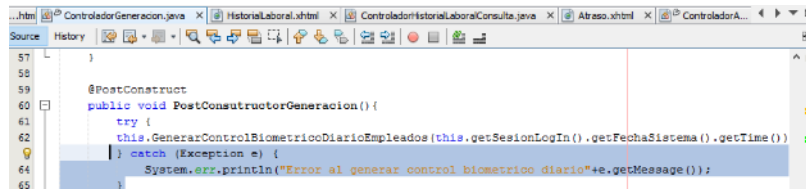
3.1) Error al intentar insertar registros externos a locales



```
try {
    negocioRegistro.guardar(registro);
}
//Invocar calculo de asistencia y demas mediante el motor de generacion
NegocioMotorRegistro motorRegistro=new NegocioMotorRegistro(this.getSessionLogIn());
motorRegistro.generacionControlHistorialPeriodo(registroExternos.get(0).getDiaRegistro().getDia());
}
else{
    msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin registros empleados Externo", "Emp
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
}
} catch (Exception e) {
    System.err.println("Error al intentar "+e.getMessage());
}
```

Figura 38-3: Error al intentar insertar registros externos a locales
Realizado por: Erika R. 2017

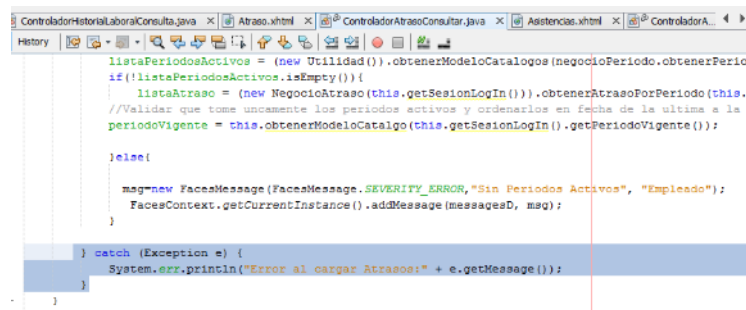
3.2) Error al generar control biométrico vacaciones



```
@PostConstruct
public void PostConstructorGeneracion(){
    try {
        this.GenerarControlBiotmetricoDiarioEmpleados(this.getSessionLogIn().getFechaSistema().getTime());
    } catch (Exception e) {
        System.err.println("Error al generar control biometrico diario"+e.getMessage());
    }
}
```

Figura 39-3: Error al generar control biométrico vacaciones
Realizado por: Erika R. 2017

3.3) Error al intentar cargar lista de atrasos generados



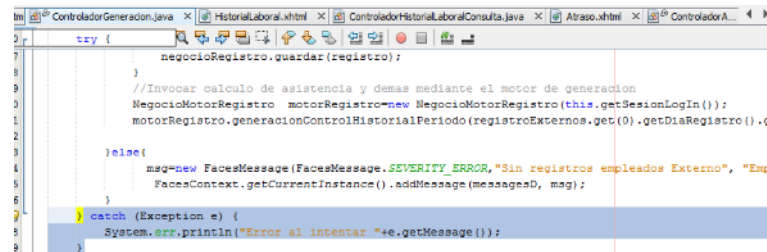
```
listaPeriodosActivos = (new Utilidad()).obtenerModeloCatalogos(negocioPeriodo.obtenerPeriodosActivos());
if(!listaPeriodosActivos.isEmpty()){
    listaAtraso = (new NegocioAtraso(this.getSessionLogIn())).obtenerAtrasoPorPeriodo(this.getSessionLogIn(), listaPeriodosActivos);
    //Validar que tome unicamente los periodos activos y ordenarlos en fecha de la ultima a la primera
    periodoVigente = this.obtenerModeloCatalogo(this.getSessionLogIn().getPeriodoVigente());
}
else{
    msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin Periodos Activos", "Empleado");
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
}
} catch (Exception e) {
    System.err.println("Error al cargar Atrasos:" + e.getMessage());
}
```

Figura 40-3: Error al intentar cargar lista de atrasos generados
Realizado por: Erika R. 2017

4) Generar asistencia a partir del registro diario

Total, errores no controlados: 3

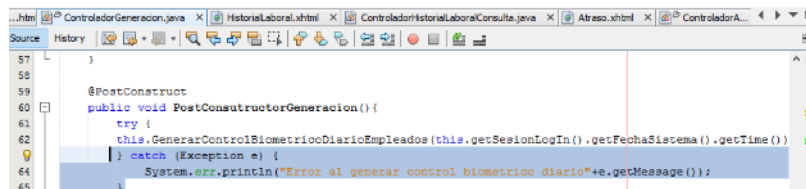
4.1) Error al intentar insertar registros externos a locales



```
try {
    negocioRegistro.guardar(registro);
}
//Invocar calculo de asistencia y demas mediante el motor de generacion
NegocioMotorRegistro motorRegistro=new NegocioMotorRegistro(this.getSessionLogIn());
motorRegistro.generacionControlHistorialPeriodo(registroExternos.get(0).getDiaRegistro());
} else {
    msg=new FacesMessage(FacesMessage.SEVERITY_ERROR,"Sin registros empleados Externo", "Emg");
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
}
} catch (Exception e) {
    System.err.println("Error al intentar insertar registros externos a locales");
}
```

Figura 41-3: Error al intentar insertar registros externos a locales, asistencia
Realizado por: Erika R. 2017

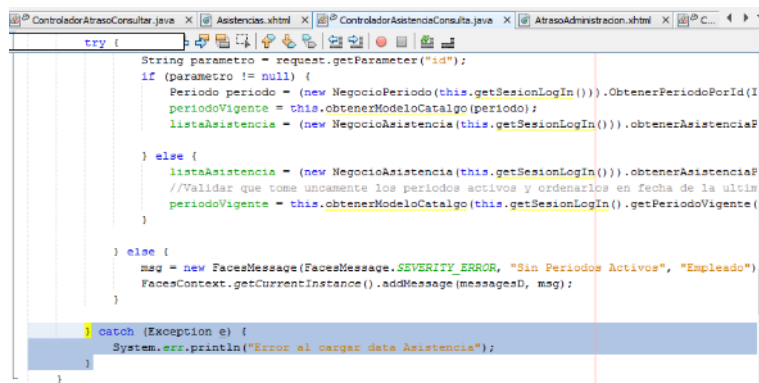
4.2) Error al generar control biométrico vacaciones



```
@PostConstruct
public void PostConstructorGeneracion() {
    try {
        this.GenerarControlBiotriicoDiarioEmpleados(this.getSessionLogIn().getFechaSistema().getTime());
    } catch (Exception e) {
        System.err.println("Error al generar control biometrico diario");
    }
}
```

Figura 42-3: Error al intentar insertar registros externos a locales, vacaciones
Realizado por: Erika R. 2017

4.3) Error al intentar cargar lista de asistencias generadas



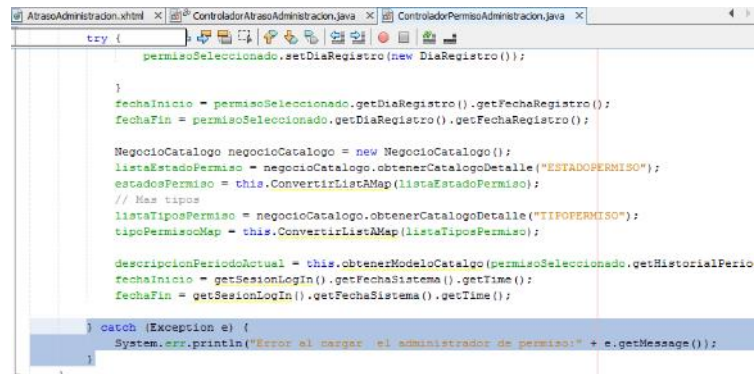
```
try {
    String parametro = request.getParameter("id");
    if (parametro != null) {
        Periodo periodo = (new NegocioPeriodo(this.getSessionLogIn())).ObtenerPeriodoPorId(I);
        periodoVigente = this.obtenerModeloCatalogo(periodo);
        listaAsistencia = (new NegocioAsistencia(this.getSessionLogIn())).obtenerAsistenciaP
    } else {
        listaAsistencia = (new NegocioAsistencia(this.getSessionLogIn())).obtenerAsistenciaP
        //Validar que tome unicamente los periodos activos y ordenarlos en fecha de la ultim
        periodoVigente = this.obtenerModeloCatalogo(this.getSessionLogIn().getPeriodoVigente(
    }
} else {
    msg = new FacesMessage(FacesMessage.SEVERITY_ERROR, "Sin Periodos Activos", "Empleado");
    FacesContext.getCurrentInstance().addMessage(messagesD, msg);
}
} catch (Exception e) {
    System.err.println("Error al cargar lista de asistencias generadas");
}
```

Figura 43-3: Error al intentar cargar lista de asistencias generadas
Realizado por: Erika R. 2017

5) Registrar permisos por tipo

Total, errores no controlados: 4

5.1) Error al cargar el administrador de permisos



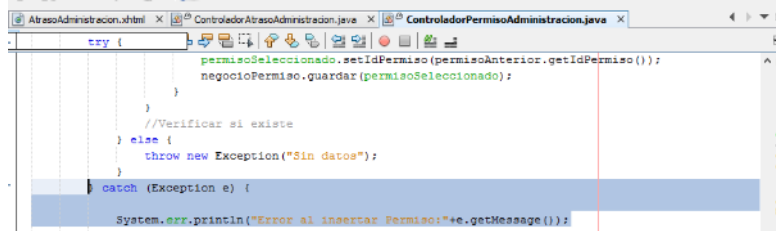
```
try {
    permisoSeleccionado.setDiaRegistro(new DiaRegistro());
}
fechaInicio = permisoSeleccionado.getDiaRegistro().getFechaRegistro();
fechaFin = permisoSeleccionado.getDiaRegistro().getFechaRegistro();

NegocioCatalogo negocioCatalogo = new NegocioCatalogo();
listaEstadoPermiso = negocioCatalogo.obtenerCatalogoDetalle("ESTADOPERMISO");
estadosPermiso = this.ConvertirListAMap(listaEstadoPermiso);
// Mas tipos
listaTiposPermiso = negocioCatalogo.obtenerCatalogoDetalle("TIPOPERMISO");
tipoPermisoMap = this.ConvertirListAMap(listaTiposPermiso);

descripcionPeriodoActual = this.obtenerModeloCatalogo(permisoSeleccionado.getHistoriaPermisos());
fechaInicio = getSessionLogin().getFechaSistema().getTime();
fechaFin = getSessionLogin().getFechaSistema().getTime();
} catch (Exception e) {
    System.err.println("Error al cargar el administrador de permisos:" + e.getMessage());
}
```

Figura 44-3: Error al cargar el administrador de permisos
Realizado por: Erika R. 2017

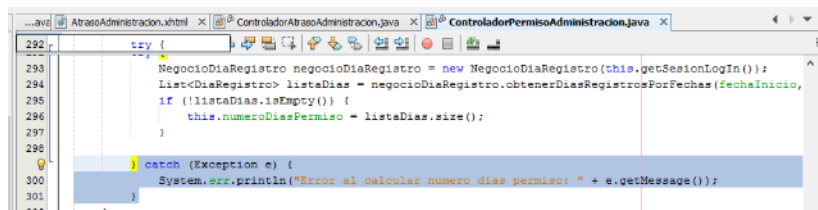
5.2) Error al guardar el permiso por tipo



```
try {
    permisoSeleccionado.setIdPermiso(permisoAnterior.getIdPermiso());
    negocioPermiso.guardar(permisoSeleccionado);
}
//Verificar si existe
} else {
    throw new Exception("Sin datos");
}
} catch (Exception e) {
    System.err.println("Error al insertar Permiso:" + e.getMessage());
}
```

Figura 45-3: Error al guardar el permiso por tipo
Realizado por: Erika R. 2017

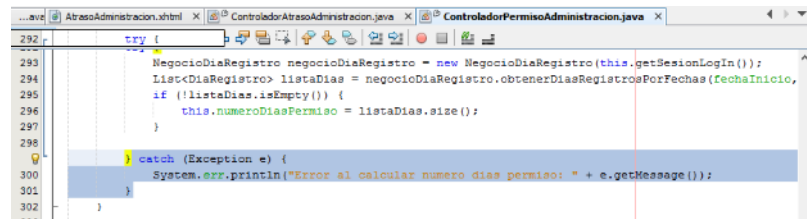
5.3) Error al calcular número de días de permiso



```
try {
    NegocioDiaRegistro negocioDiaRegistro = new NegocioDiaRegistro(this.getSessionLogin());
    List<DiaRegistro> listaDias = negocioDiaRegistro.obtenerDiasRegistrosPorFechas(fechaInicio, fechaFin);
    if (!listaDias.isEmpty()) {
        this.numeroDiasPermiso = listaDias.size();
    }
} catch (Exception e) {
    System.err.println("Error al calcular numero dias permiso: " + e.getMessage());
}
```

Figura 46-3: Error al calcular número de días de permiso
Realizado por: Erika R. 2017

5.4) Error al obtener datos del empleado



```
...avz | AbrasoAdministracion.xhtml | ControladorAtrasoAdministracion.java | ControladorPermisoAdministracion.java
292 | try {
293 |     NegocioDiaRegistro negocioDiaRegistro = new NegocioDiaRegistro(this.getSessionLogIn());
294 |     List<DiaRegistro> listaDias = negocioDiaRegistro.obtenerDiasRegistrosPorFechas(fechaInicio,
295 |     if (!listaDias.isEmpty()) {
296 |         this.numeroDiasPermiso = listaDias.size();
297 |     }
298 | } catch (Exception e) {
299 |     System.err.println("Error al calcular numero dias permiso: " + e.getMessage());
300 | }
301 |
302 | }
```

Figura 47-3: Error al obtener datos del empleado
Realizado por: Erika R. 2017

Característica: Eficiencia interna

Subcaracterística: Comportamiento temporal.

Métrica: Tiempo de respuesta

Tabla 61-3: Resultados métrica interna tiempo de respuesta

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y de cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Tiempo de respuesta	¿Cuál es el tiempo estimado para completar una tarea?	Evaluar la eficiencia de las llamadas al SO y a la aplicación. Estimar el tiempo de respuesta basado en ello.	<ol style="list-style-type: none"> 1. X=2 seg. 2. X=4 seg. 3. X=3 seg. 4. X=3 seg. 5. X=2 seg. 	2 seg.	Proporción	X=tiempo	Sistema Operativo conocido. Tiempo estimado en llamadas al sistema.	Usuarios desarrolladores

Realizado por: Erika Reina. 2017

Fuente: (Mena, 2006: p.13)

Característica: Eficiencia externa

Subcaracterística: Comportamiento temporal.

Métrica: Tiempo de respuesta

Tabla 62-3: Resultados Métrica Externa Tiempo de Respuesta

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Tiempo de respuesta	¿Cuánto tiempo le ha tomado terminar una tarea específica? ¿Cuánto tiempo le toma recibir una respuesta a las tareas específicas?	Empiece una tarea especificada. Mida el tiempo que toma para la muestra para terminar su operación. Guarde un registro de cada intento.	<ol style="list-style-type: none"> 1. X= 3,093 s 2. X= 3,833 s 3. X= 6,112 s 4. X= 6,117 s 5. X= 0,525 s 	0,525 s	Ratio	X=tiempo	Reporte de prueba. Informe de la operación mostrada en un lapso de tiempo	Usuarios desarrolladores

Realizado por: Erika Reina. 2017

Fuente: (Vivanco Villamar, 2011: p.123)

1. Registrar datos empleados

Tiempo 3093 ms, registrar datos de un empleado

IDENTIFICACION	PRIMER NOMBRE	SEGUNDO NOMBRE	APELLIDO PATERNO	APELLIDO MATERNO	GENERO	FECHA NACIMIENTO	ESTADO CIVIL
030344444	YYYYYY	QQQ		U	F		SOLTERO
0804344661	MELANIE	AMANDA	RAMON	GUÑA	F	2016-01-14	SOLTERO
0000000001	LUIS	ALBERTO	MORALES	PEREZ		1992-01-10	
323232	R2323	RWERWE					
039404949	LUIS	GUSTAVO	PATIÑO	CASTILLO	M	1998-01-03	VIUDO
39393393	FDSFSD	FDSFDS	FDSFDS	FDSFSD	M	2000-01-18	SOLTERO
0000000002	MARIA	JOSEFINA	FERNANDEZ	LOPEZ		1991-01-10	

Summary	Count	Count
Played	1	Assertions 0
Succeeded	1	Warnings 2
Failed	0	Timeouts 0
Avg Time (ms)	3093	Max Time (ms) 3093

Figura 48-3: Tiempo de registro datos de un empleado
Realizado por: Erika R. 2017

Total, tiempo de registro datos empleado.

Total, tiempo: 3,093 s.

Para la medición de los tiempos de las tareas 2, 3 y 4 se realizó primero la medición del registro diario de los empleados ya que a través de este proceso se generan los antes mencionados. Posterior a ello se muestra el tiempo que tardan en visualizarse cada una de las tareas 2, 3 y 4 individualmente.

Registro diario.

- Tiempo: 1015 ms, Autenticación

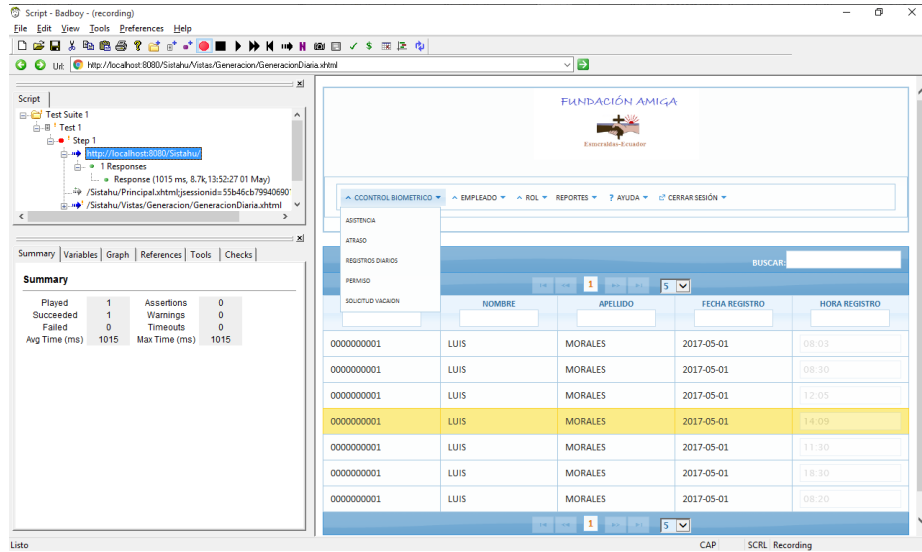


Figura 49-3: Tiempo de registro diario
Realizado por: Erika R. 2017

Tiempo: 1146 ms, Abarca los procesos de traslado de información desde el lector de huellas digitales hacia el sistema local y la generación de días de vacaciones, asistencia y atrasos.

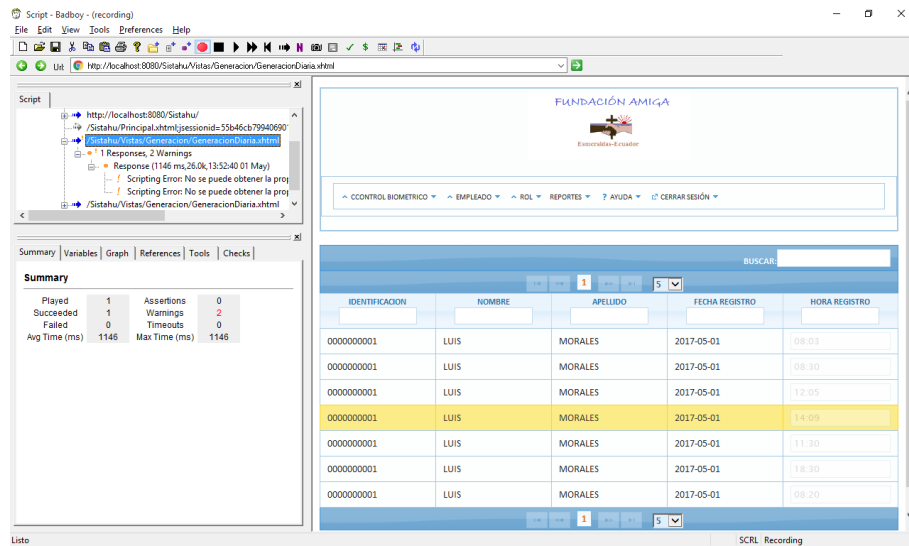


Figura 50-3: Tiempo de traslado de información y generación de vacaciones, asistencia y atrasos.
Realizado por: Erika R. 2017

Tiempo: 1066 ms, visualización de registros diarios

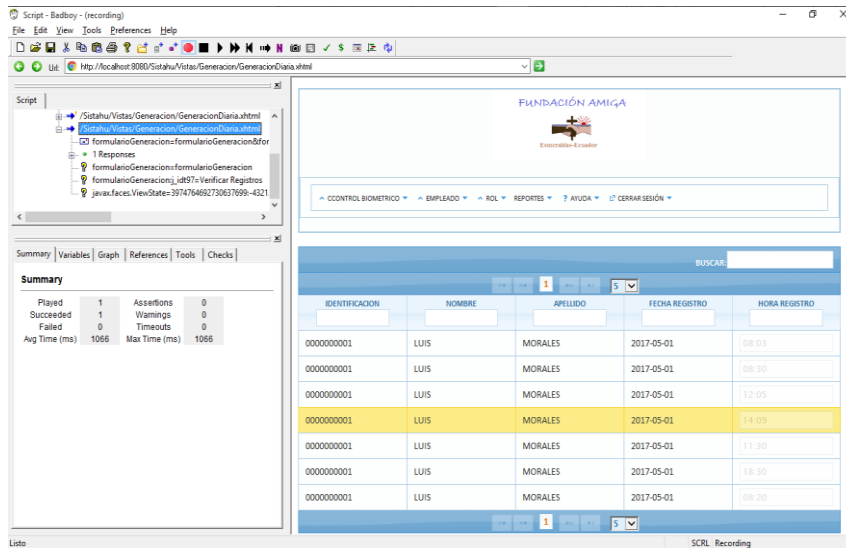


Figura 51-3: Tiempo de visualización de registros diarios.
Realizado por: Erika R. 2017

Tiempo total registros diarios: $1015+1146+1066 = 3227$ ms

Tiempo total registros diarios: 3,227 s

6. Generar los días de vacaciones a partir del registro diario

Tiempo 606 ms, visualización días vacaciones generadas

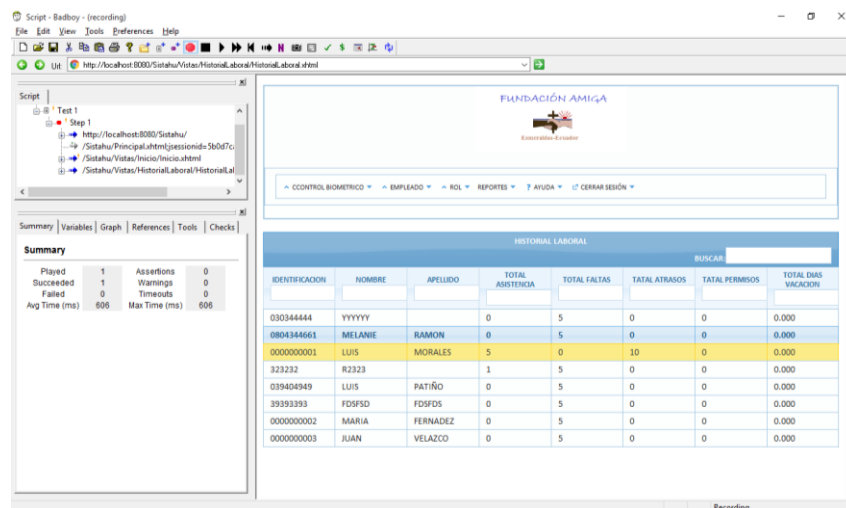


Figura 52-3: Tiempo de visualización días vacaciones generadas.
Realizado por: Erika R. 2017

Total, tiempo generación días vacación es igual al total de tiempo obtenido en la transacción previa sumado al tiempo de la visualización de la transacción actual.

Total, tiempo= $3227 + 606 = 3833$ ms

Total, tiempo= 3,833 s

7. Generar atrasos a partir del registro diario

Tiempo 2885 ms, visualización de atrasos generados.

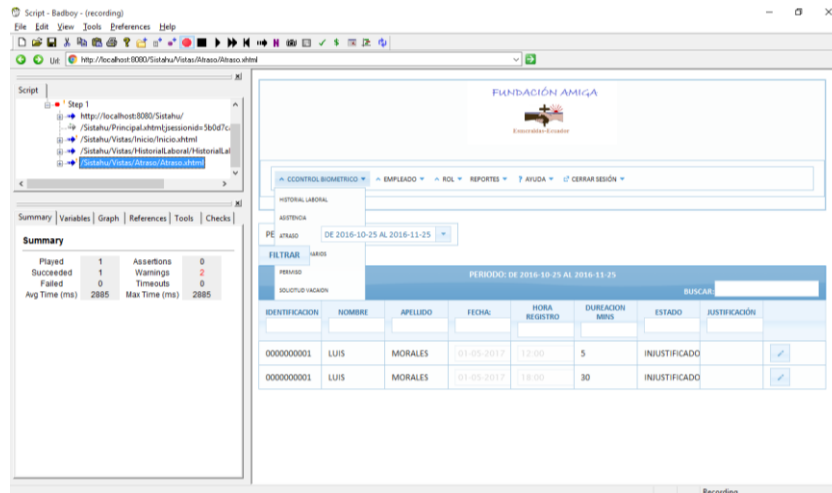


Figura 53-3: Tiempo de visualización de atrasos generados.
Realizado por: Erika R. 2017

Total, tiempo generación atrasos es igual al total de tiempo obtenido en la transacción de registro diario sumado al tiempo de la visualización de la transacción actual.

Total, Tiempo= 3227 ms + 2885 ms = 6112 ms

Total, Tiempo= 6,112 s

8. Generar asistencia a partir del registro diario

Tiempo 2890 ms, visualización de asistencias generados.

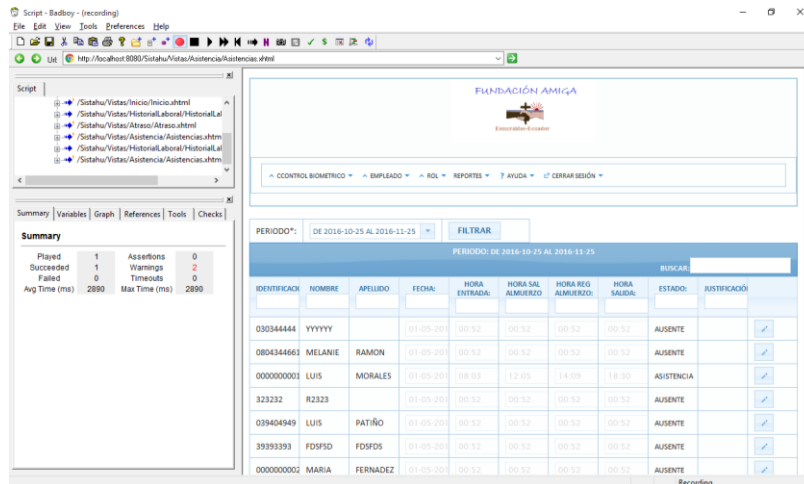


Figura 54-3: Tiempo de visualización de asistencias generados.
Realizado por: Erika R. 2017

Total, tiempo generación atrasos es igual al total de tiempo obtenido en la transacción de registro diario sumado al tiempo de la visualización de la transacción actual.

Total, tiempo= 3227 ms +2890 ms = 6117 ms

Total, tiempo= 6,117 s

9. Registrar permisos por tipo

Tiempo 525 ms, registrar un permiso por tipo

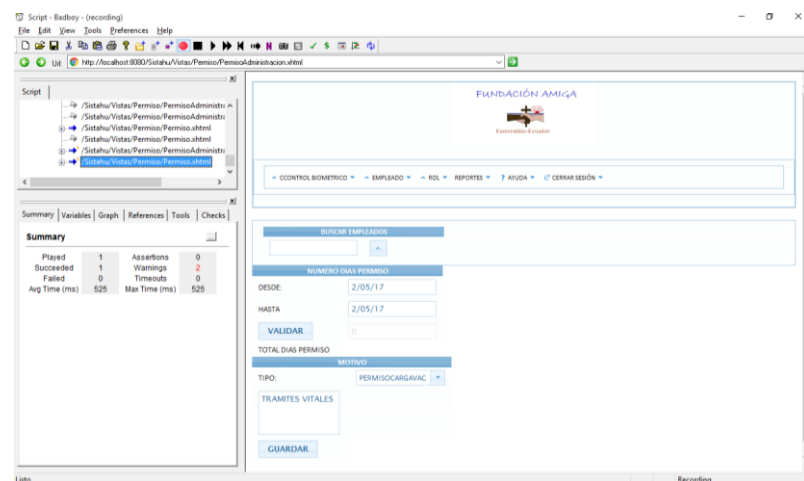


Figura 55-3: Tiempo de registrar un permiso por tipo.
Realizado por: Erika R. 2017

Total, tiempo de registro de un permiso por tipo.

Total, tiempo= 525 ms

Total, tiempo= 0,525 s

Característica: Eficiencia externa

Subcaracterística: Utilización de recursos.

Métrica: Uso de memoria ram

Tabla 63-3: Resultados Métrica Externa Uso de Memoria Ram

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Uso de Memoria	¿Cuánta memoria ha utilizado para terminar una tarea específica?	Empiece una tarea especificada. Mida la cantidad de memoria que toma para terminar su operación. Guarde un registro de cada intento.	<ol style="list-style-type: none"> 1. X= 141910.35 MB 2. X= 111478.57 MB 3. X= 138934.13 MB 4. X= 128750.40 MB 5. X= 78766.70 MB 	78766.70 MB	Ratio	X= Mega Bytes	Reporte de prueba. Informe de la operación mostrada en capacidad de memoria usada	Usuarios desarrolladores

Realizado por: Erika Reina. 2017

Fuente: (Vivanco Villamar, 2011: p.112)

Tamaño en MB que ocupa la aplicación en memoria RAM al realizar la tarea especificada.

1) Registrar datos del empleado

Memoria promedio usada: 145.316.208 Bytes → 145.316 .208 Bytes /1024 MB → 141910.35 MB

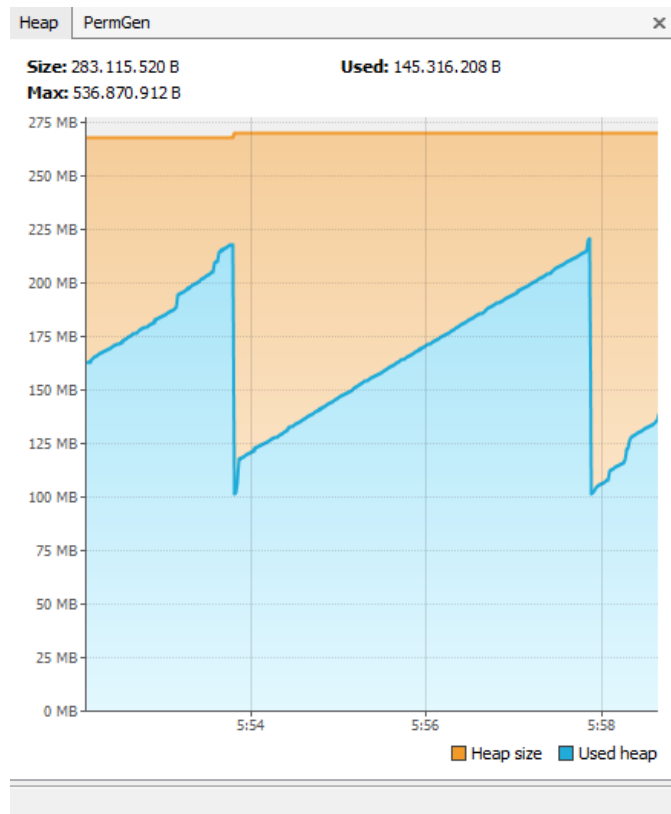


Figura 56-3: Memoria usada registro datos del empleado
Realizado por: Erika R. 2017

2) Generar los días de vacaciones a partir del registro diario

Memoria promedio usada: 114.154 .056 Bytes → 114.154 .056 Bytes /1024 → 111478.57 MB

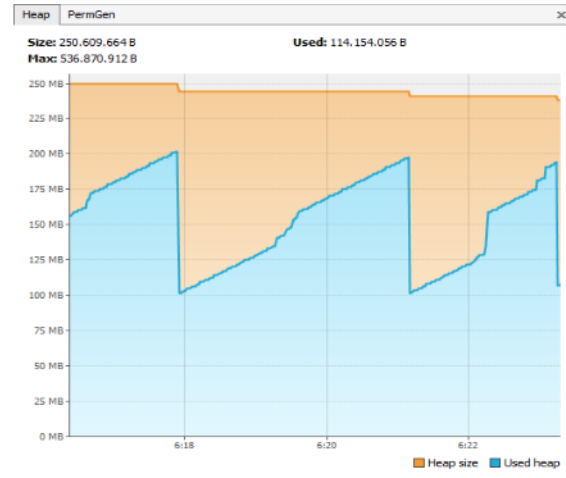


Figura 57-3: Memoria usada en generación los días de vacaciones a partir del registro diario
 Realizado por: Erika R. 2017

3) Generar atrasos a partir del registro diario

Memoria promedio usada: 142.268 .552 Bytes → 142.268 .552 Bytes /1024 MB → 138934.13 MB

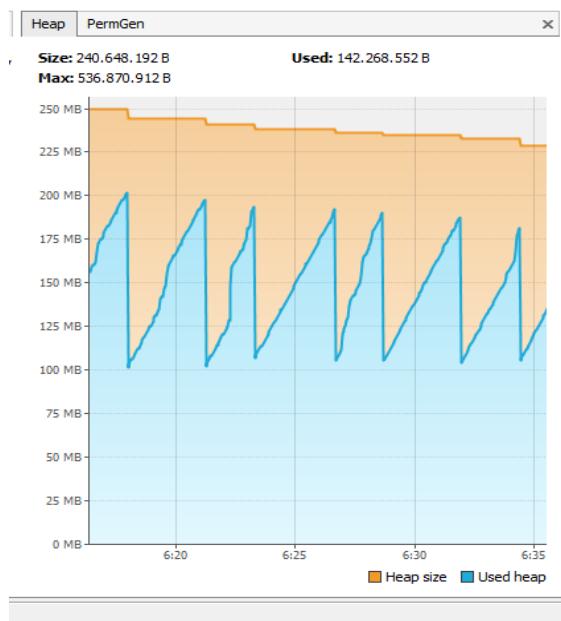


Figura 58-3: Memoria usada en atrasos a partir del registro diario
 Realizado por: Erika R. 2017

4) Generar asistencia a partir del registro diario

Memoria promedio usada: 131.840 .408 Bytes → 131.840 .408 Bytes /1024 MB → 128750.40 MB

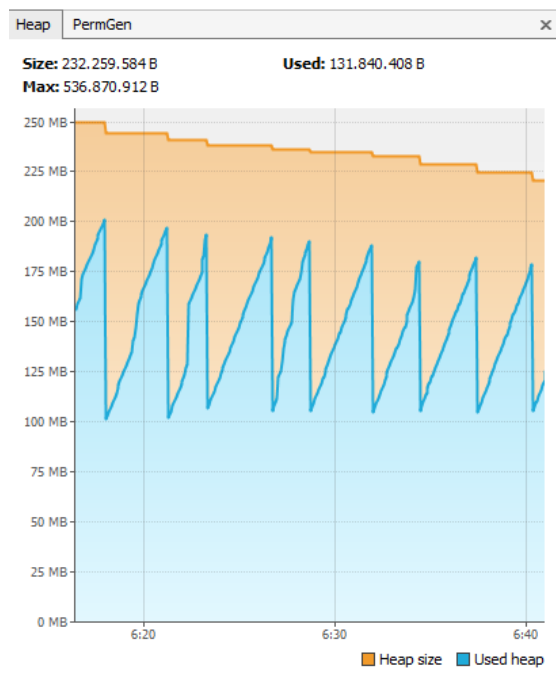


Figura 59-3: Memoria usada en asistencia a partir del registro diario
 Realizado por: Erika R. 2017

5) Registrar permisos por tipo

Memoria promedio usada: 80.657 .096 Bytes → 80.657.096 Bytes /1024 MB → 78766.70 MB

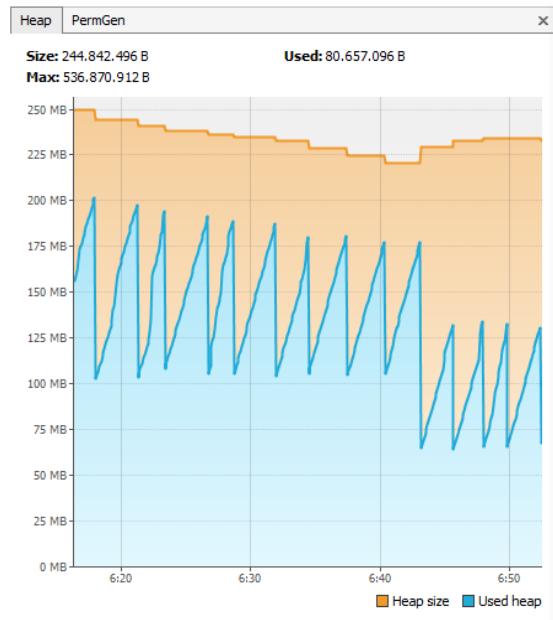


Figura 60-3: Memoria usada en registro permisos por tipo
 Realizado por: Erika R. 2017

Característica: Eficiencia Externa

Subcaracterística: Utilización de Recursos.

Métrica: Uso de Procesador

Tabla 64-3: Resultados Métrica Externa Uso de Procesador

Nombre de la métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculo de datos	Interpretación de los valores medidos	Tipo de escala métrica	Tipo de medida	Entradas para medición	Usuarios seleccionados
Uso de Procesador	¿Cuánto procesador ha utilizado para terminar una tarea específica?	Empiece una tarea especificada. Mida la cantidad de procesador que toma para terminar su operación. Guarde un registro de cada intento.	<ol style="list-style-type: none"> 1. X= 9.3% 2. X= 4.2% 3. X= 5.8% 4. X= 3.4% 5. X= 5.0% 	3.4%	Ratio	X= Porcentaje de Uso	Reporte de prueba. Informe de la operación mostrada en porcentaje de uso	Usuarios desarrolladores

Realizado por: Erika Reina. 2017

Fuente: (Vivanco Villamar, 2011: p.111)

Porcentaje promedio de uso del procesador mientras se realiza la tarea específica.

1) Registrar datos del empleado

Porcentaje promedio uso procesador: 9.3%

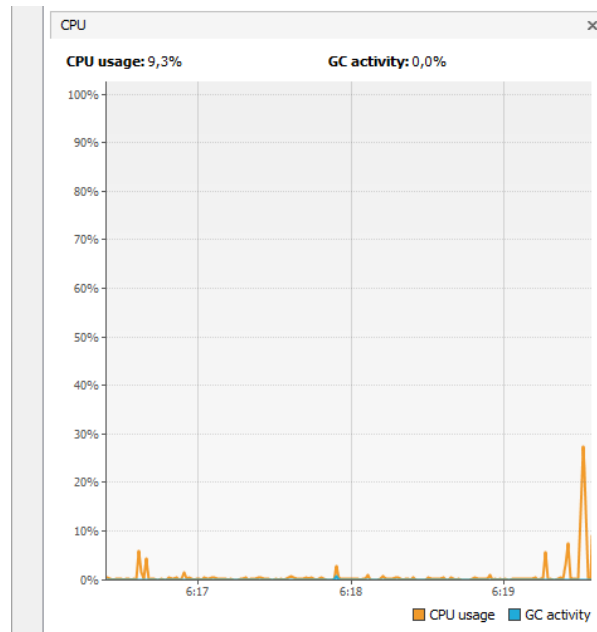


Figura 61-3: Procesador usado en registro datos del empleado
Realizado por: Erika R. 2017

2) Generar los días de vacaciones a partir del registro diario

Porcentaje promedio uso procesador: 4.2%

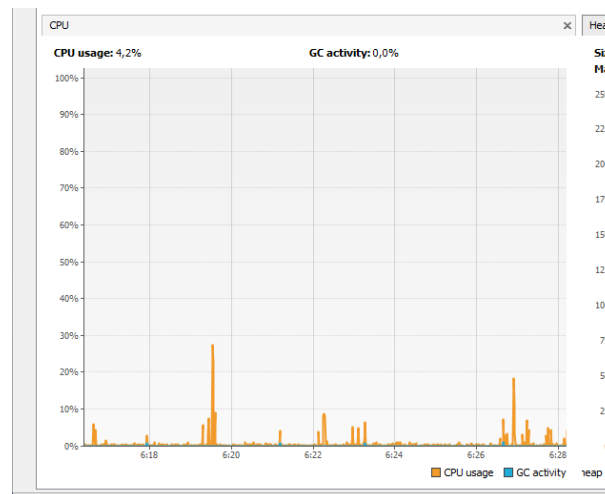


Figura 62-3: Procesador usado en generación de vacaciones a partir del registro diario
Realizado por: Erika R. 2017

3) Generar atrasos a partir del registro diario

Porcentaje promedio uso procesador: 5.8%

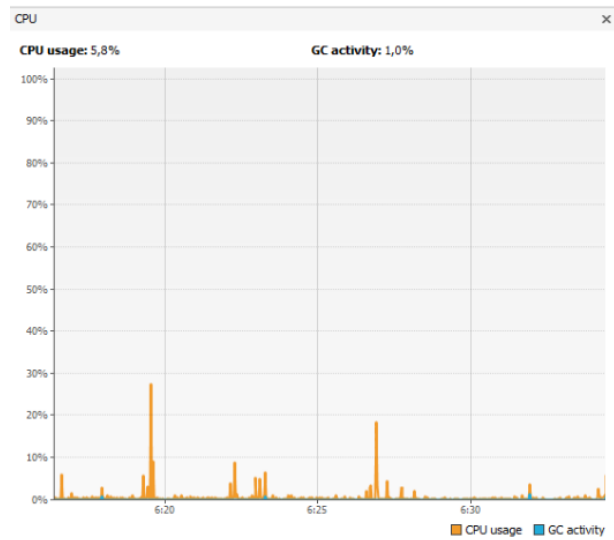


Figura 63-3: Procesador usado en generación de atrasos a partir del registro diario

Realizado por: Erika R. 2017

4) Generar asistencia a partir del registro diario

Porcentaje promedio uso procesador: 3.4%

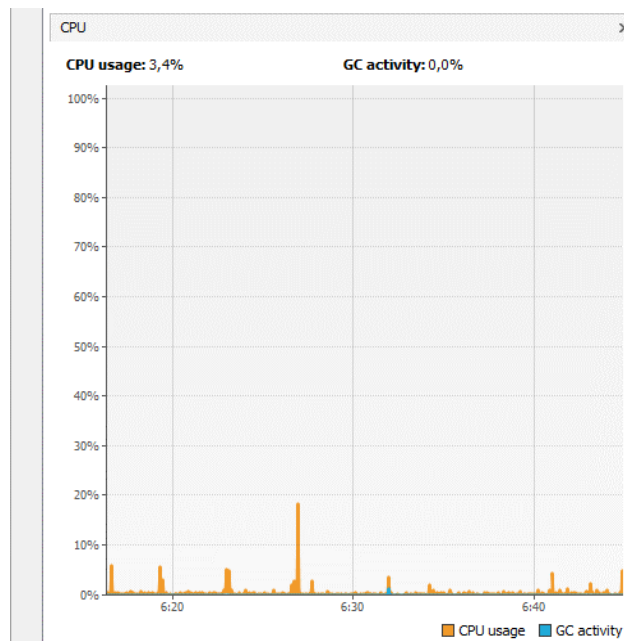


Figura 64-3: Procesador usado en generación de asistencia a partir del registro diario

Realizado por: Erika R. 2017

5) Registrar permisos por tipo

Porcentaje promedio uso procesador: 5.0%

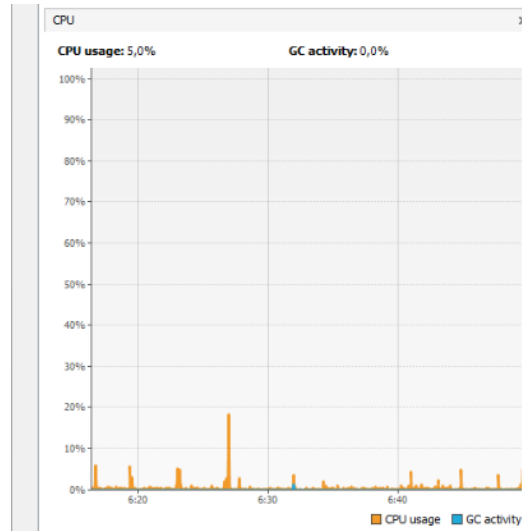


Figura 65-3: Procesador usado en registro permisos por tipo
Realizado por: Erika R. 2017

Resultados obtenidos de la medición de las métricas

En la **tabla 64-3** se puede visualizar la ponderación realizada a las características interna y externa del software tomando como primordial a la utilización de recursos de la parte externa del software debido a que el uso tanto de la memoria como del procesador con un 45% de importancia influyeron en gran medida a la generación de una respuesta inmediata del sistema con un 25%. Sin embargo, es necesario mencionar que se mantuvo un equilibrio en cuanto al tiempo de respuesta de tareas internas ya que se determinó un 20 % de importancia el mismo que se veía reflejado en las densidades de los mensajes; controlados y no controlados de la aplicación.

Tabla 65-3: Ponderación de las métricas de eficiencia interna y externa del SISTAHU

Características	Subcaracterísticas	Nivel de Importancia	Ponderación
Interna	Utilización de Recursos	Opcional	10 %
	Comportamiento Temporal	Primordial	20 %
Externa	Utilización de Recursos	Primordial	45 %
	Comportamiento Temporal	Primordial	25 %

Realizado por: Erika Reina. 2017

En el **gráfico 9-3** se determinó la ponderación para la eficiencia del SISTAHU teniendo como de mayor importancia a la eficiencia externa con un 70% y un 30% a la eficiencia interna. Esta ponderación se la realizó tomando en cuenta el grado de importancia que el usuario en conjunto con el desarrollador le asignó a cada una de las subcaracterísticas y métricas comprendidas en la norma IS/IEC 9126-2 y 9126-3.

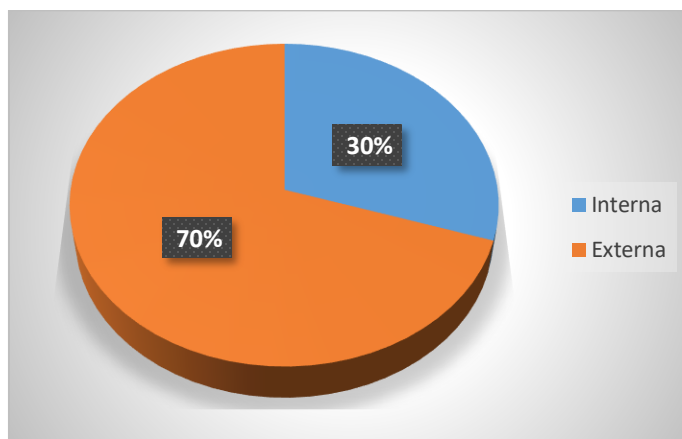


Gráfico 9-3: Gráfico de la ponderación para la eficiencia del SISTAHU
Realizado por: Erika R. 2017

Determinación de la Eficiencia de SISTAHU

Para la determinación de la eficiencia tanto interna como externa se tomará en cuenta los siguientes valores para ubicar los datos obtenidos de las mediciones y establecer si cumplen con el grado de eficiencia propuesto en la norma seleccionada.

Tabla 66-3: Criterios de valoración para determinación de eficiencia de SISTAHU

Escala de medición	Niveles de puntuación	Grado de satisfacción
0-2.75	Inaceptable	Insatisfactorio
2.76-5	Mínimamente aceptable	
5.01-8.75	Rango objetivo	Satisfactorio
8.76-10	Excede los Requisitos	Muy Satisfactorio

Realizado por: Vivanco Villamar. 2011

Fuente: (Vivanco Villamar, 2011: p.149)

Eficiencia interna

Subcaracterística: Utilización de recursos

Métrica: Utilización I/O densidad de mensaje

Para este fin se empleó el registro de valoración de cada uno de los requisitos medidos para la subcaracterística especificada en una escala de 0 a 10 tomando al mayor como el de valor 10.

Tabla 67-3: Valores obtenidos de utilización I/O densidad de mensaje

Requerimiento	Utilización I/O Densidad de mensaje	Valor de medición
Ingresar de datos del empleado	0.0069	10.00
Generar los días de vacaciones a partir del registro diario	0.0020	2.89
Generar atrasos mediante el registro diario del empleado	0.0036	5.22
Generar asistencia a partir del registro diario del empleado	0.0032	4.64
Registrar permisos por tipo	0.0052	7.54
Promedio		6.06

Realizado por: Erika R. 2017

Al analizar la característica eficiencia interna con la subcaracterística utilización I/O densidad de mensaje se obtuvo un promedio de los datos obtenidos de 6.06, pertenecientes a los requerimientos evaluados, el cual está en la escala de 5 a 8.75 lo que nos determina que se encuentra en el rango objetivo cumpliendo satisfactoriamente la eficiencia interna.

Subcaracterística: Utilización de recursos

Métrica: Utilización de memoria densidad de mensaje

Para este fin se empleó el registro de valoración de cada uno de los requisitos medidos para la Subcaracterística especificada en una escala de 0 a 10 tomando al mayor como el de valor 10.

Tabla 68-3: Valores obtenidos de utilización de memoria densidad de mensaje

Requerimiento	Utilización de Memoria Densidad de Mensaje	Valor de medición
Ingresar de datos del empleado	0.0086	4.11
Generar los días de	0.0059	2.82

vacaciones a partir del registro diario		
Generar atrasos mediante el registro diario del empleado	0.0054	2.58
Generar asistencia a partir del registro diario del empleado	0.0048	2.30
Registrar permisos por tipo	0.0209	10.00
Promedio		4.36

Realizado por: Erika R. 2017

Al analizar la característica eficiencia interna con la subcaracterística utilización de memoria densidad de mensaje se obtuvo un promedio de los datos obtenidos de 4.36, pertenecientes a los requerimientos evaluados, el cual está en la escala de 2.75 a 5 lo que nos determina que se encuentra en el rango mínimamente aceptable sin embargo tiene un grado de satisfacción insatisfactorio lo que significa que no cumple con el valor adecuado para ser eficiente según la norma ISO/IEC 9126-3.

Subcaracterística: Comportamiento temporal

Métrica: Tiempo de respuesta

Para este fin se empleó el registro de valoración de cada uno de los requisitos medidos para la Subcaracterística especificada en una escala de 0 a 10 tomando al mayor como el de valor 10.

Tabla 69-3: Valores obtenidos de Tiempo de respuesta interno

Requerimiento	Tiempo de Respuesta	Valor de medición
Ingresar de datos del empleado	2 s	5.00
Generar los días de vacaciones a partir del registro diario	4 s	10.00
Generar atrasos mediante el registro diario del empleado	3 s	7.50
Generar asistencia a partir del registro diario del empleado	3 s	7.50
Registrar permisos por tipo	2 s	5.00
Promedio		7.00

Realizado por: Erika R. 2017

Al analizar la característica eficiencia interna con la métrica tiempo de respuesta se obtuvo un promedio de los datos obtenidos de 7 s, pertenecientes a los requerimientos evaluados, el cual está en la escala de 5 a 8.75 lo que nos determina que se encuentra en el rango objetivo cumpliendo satisfactoriamente la eficiencia interna.

En el **gráfico 10-3** podemos visualizar el porcentaje que tienen las métricas implicadas en determinar la eficiencia interna.

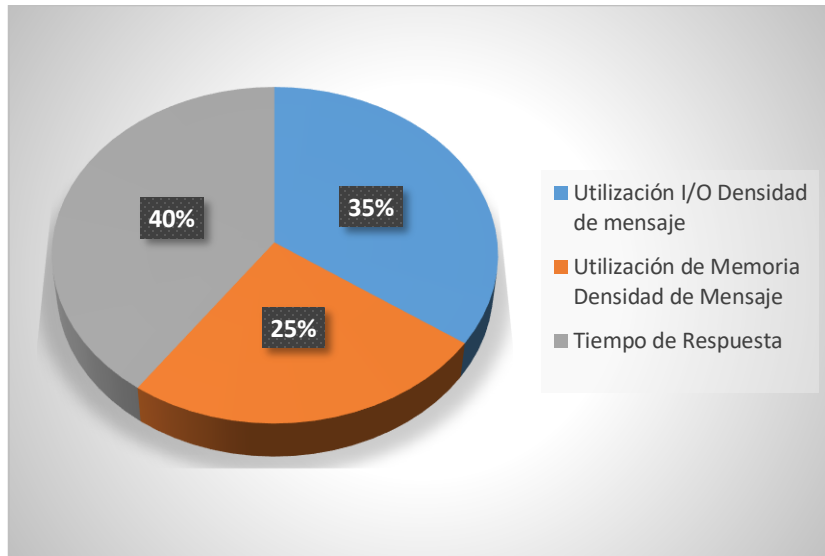


Gráfico 10-3: Gráfico de las métricas de la eficiencia interna del SISTAHU
Realizado por: Erika R. 2017

Eficiencia externa

Subcaracterística: Utilización de recursos

Métrica: Uso de memoria

Para este fin se empleó el registro de valoración de cada uno de los requisitos medidos para la subcaracterística especificada en una escala de 0 a 10 tomando al mayor como el de valor 10.

Tabla 70-3: Valores obtenidos de uso de memoria ram

Requerimiento	Uso de Memoria	Valor de medición
Ingresar de datos del empleado	141910.35 mb	10.00
Generar los días de vacaciones a partir del registro diario	111478.57 mb	7.86
Generar atrasos mediante el registro diario del empleado	138934.13 mb	9.79
Generar asistencia a partir del registro diario del empleado	128750.40 mb	9.07
Registrar permisos por tipo	78766.70 mb	5.55
Promedio		8.45

Realizado por: Erika R. 2017

Al analizar la característica eficiencia externa con la métrica uso de memoria se obtuvo un promedio de los datos obtenidos de 8.45 mb, pertenecientes a los requerimientos evaluados, el cual está en la escala de 5 a 8.75 lo que nos determina que se encuentra en el rango objetivo cumpliendo satisfactoriamente la eficiencia externa.

Subcaracterística: Utilización de recursos

Métrica: Uso de procesador

Para este fin se empleó el registro de valoración de cada uno de los requisitos medidos para la subcaracterística especificada en una escala de 0 a 10 tomando al mayor como el de valor 10.

Tabla 71-3: Valores obtenidos de uso de procesador

Requerimiento	Uso de Procesador	Valor de medición
Ingresar de datos del empleado	9.3 %	10.00
Generar los días de vacaciones a partir del registro diario	4.2 %	4.51
Generar atrasos mediante el registro diario del empleado	5.8 %	6.24
Generar asistencia a partir del registro diario del empleado	3.4%	3.66
Registrar permisos por tipo	5.0 %	5.37
Promedio		5.96

Realizado por: Erika R. 2017

Al analizar la característica eficiencia externa con la métrica uso de procesador se obtuvo un promedio de los datos obtenidos de 5.96, pertenecientes a los requerimientos evaluados, el cual está en la escala de 5 a 8.75 lo que nos determina que se encuentra en el rango objetivo cumpliendo satisfactoriamente la eficiencia externa.

Subcaracterística: Comportamiento temporal

Métrica: Tiempo de respuesta

Para este fin se empleó el registro de valoración de cada uno de los requisitos medidos para la subcaracterística especificada en una escala de 0 a 10 tomando al mayor como el de valor 10.

Tabla 72-3: Valores obtenidos de tiempo de respuesta externo

Requerimiento	Tiempo de respuesta	Valor de medición
Ingresar de datos del empleado	3.093 s	5.06 s
Generar los días de vacaciones a partir del registro diario	3.833 s	6.27 s
Generar atrasos mediante el registro diario del empleado	6.112 s	9.99 s
Generar asistencia a partir del registro diario del empleado	6.117 s	10 s
Registrar permisos por tipo	0.525 s	0.86 s
Promedio		6.44 s

Realizado por: Erika R. 2017

En el **gráfico 11-3** podemos visualizar el porcentaje que tienen las métricas implicadas en determinar la eficiencia interna.

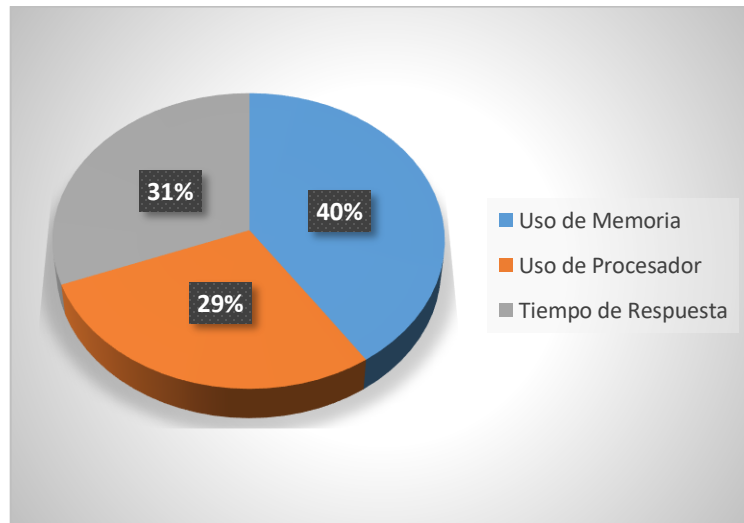


Gráfico 11-3: Gráfico de las métricas de la eficiencia externa del SISTAHU
Realizado por: Erika R. 2017

Resumen de los valores obtenidos en las métricas de eficiencia interna y externa, así como de la ponderación realiza para las subcaracterísticas especificadas.

Tabla 73-3 Resumen de las métricas de eficiencia interna y externa

Subcaracterística	Métrica	Valor	Ponderación
Utilización de Recursos	Utilización I/O	6.06	5%
	Densidad de mensaje		
	Utilización de Memoria	4.36	5%
	Densidad de Mensaje		
Comportamiento Temporal	Tiempo de Respuesta	7.00	20 %
Utilización de Recursos	Uso de Memoria	8.45	22.5 %
	Uso de Procesador	5.96	22.5 %
Comportamiento Temporal	Tiempo de Respuesta	6.44	25 %

Realizado por: Erika R. 2017

En el **grafico 12-3** se puede visualizar la eficiencia total obtenida del resultado de la evaluación basados en las normas ISO/IEC 9126-2 y 9126-3 así como el faltante de eficiencia que se obtuvo de la métrica utilización de memoria densidad de mensaje de la subcaracterística utilización de recursos perteneciente a la eficiencia interna, debido a que este valor está en el rango mínimamente aceptable lo que da como resultado insatisfactorio para ser eficiente.



Gráfico 12-3: Gráfico de la eficiencia total obtenida mediante la norma ISO/IEC 9126-2 y 9126-3
Realizado por: Erika R. 2017

CONCLUSIONES

- Se determinó que los procesos de control de los empleados de la fundación, que se manejaban de forma manual ocasionaban pérdida de información por la cantidad de datos obtenidos diariamente además estos procesos tenían una duración excesiva para la complejidad de los mismos.
- Al analizar el proceso de control de empleados de la fundación se establecieron 38 requerimientos funcionales, los cuales se desarrollaron como historias de usuario y 4 requerimientos no funcionales los cuales contribuyen a mejorar la calidad del producto.
- Mediante la utilización de SISTAHU se pudo mejorar los procesos de control diarios de los empleados de la Fundación Amiga manteniendo la integridad de los datos y reduciendo el tiempo de los procesos.
- De acuerdo a lo establecido en el estándar ISO/IEC 9126-2 y 9126-3 se determinó una eficiencia interna de 25%, así como una eficiencia externa de 70% dando un total del 95% para el sistema de control biométrico SISTAHU, lo que permite al mismo efectividad y eficacia en su funcionamiento.

RECOMENDACIONES

- Mejorar la infraestructura existente para la operación del sistema, además de equipos biométricos de la marca Digital Personal.
- Investigar otros sistemas biométricos que se podrían implementar para el sistema de control SISTAHU, así como otros estándares de calidad que puedan ser empleados en proyectos futuros para la Fundación Amiga.
- Se puede complementar este trabajo con la determinación de otras características de calidad que forman parte del estándar ISO/IEC 9126.
- Emplear el manual de usuario para un correcto funcionamiento del sistema.

BIBLIOGRAFÍA

1. **ACOSTA HENRÍQUEZ, Giovanni Francisco, et al.** “Utilización de herramientas de código abierto para la generación de reportes en Java” [en línea]. *Revista tecnológica*, vol. 1(2008), pp.24-26. [Consulta 16 de 01 de 2017]. Disponible en <http://redicces.org.sv/jspui/handle/10972/522>.
2. **AMAP.** *Codificación de Java* [en línea], 14 de 01 de 2014. [Consulta 22 de 02 de 2017]. Disponible en <https://amap.cantabria.es/amap/bin/view/AMAP/CodificacionJava>.
3. **AMBROCIO, Gustavo de Jesús.** *Jasper Reports* [en línea], 03 de 05 de 2014, p.1. [Consulta 09 de 03 de 2017]. Disponible en <https://es.slideshare.net/GustavoDeJesusAmbrocio/jasperreports-34246150>.
4. **ARRIETA, A., et al.** *Gestión y Reconocimiento Óptico de los Puntos Característicos de Imágenes de Huellas Dactilares* [en línea]. Universidad de Salamanca, pp.1-2. [Consulta 09 de 01 de 2017]. Disponible en <http://www.lsi.us.es/~riquelme/red/Capitulos/LMD32.pdf>.
5. **AZUMA, Motoei.** “SQuaRE the next generation of the ISO/IEC 9126 and 14598 international standards series on software product quality” [en línea]. *ESCOM (European Software Control and Metrics conference)*. 2001, pp. 337-346. [Consulta 27 de 01 de 2017]. Disponible en [http://www.comp.ita.br/~cunha/download/CES32CE230-2002/ApresentacoesAlunos/Sem04/Cecilia_Silvana_Dagmar/iso9126-1-2-3-4\(Anexo2\).pdf](http://www.comp.ita.br/~cunha/download/CES32CE230-2002/ApresentacoesAlunos/Sem04/Cecilia_Silvana_Dagmar/iso9126-1-2-3-4(Anexo2).pdf).
6. **BATIOJA, Ricardo.** Control de Empleados de la Fundación Amiga. [entrev.] Erika Reina. *Entrevista Tesis*. Esmeraldas, 25 de 01 de 2017.
7. **BISSI, Wilson.** “SCRUM-Metodologia de desenvolvimento ágil” [en línea]. *Campo Digital*, vol. 2, no 1 (2007), pp.3-6. [Consulta 23 de 01 de 2017]. Disponible en <http://revistas.bvs-vet.org.br/campodigital/article/view/30944>.
8. **CAGUA VERA, Vicky Virginia.** *Métodos de Inspección de Calidad para la Implementación de Sistemas Desarrollados en Plataformas Heterogéneas y Aplicaciones cuatro capas* [en línea]. Tesis Doctoral. Guayaquil: Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Escuela de Ingeniería en Sistemas Computacionales. 2011. p.30. [Consulta 07 de 02 de 2017]. Disponible en <http://repositorio.cisc.ug.edu.ec/jspui/bitstream/123/198/1/METODOS%20DE%20INSPECCION%20DE%20CALIDAD%20-%2030S.pdf>.
9. **CARVALLO, J.; FRANCH, Xavier; QUER, Carmen.** “Calidad de componentes software” [en línea]. *Calidad del Producto y Proceso Software*, vol. 30, (2010), p.11. [Consulta 13 de 02

de 2017]. Disponible en <https://pdfs.semanticscholar.org/41f5/6f7a2005b4c390e55a1e02624b62e3e54936.pdf>

10. **CHACHA CHIMBORAZO, Jorge Giovanni; TACO, Luis Anibal.** *Desarrollo de sistema de control biométrico de docentes del Departamento de Eléctrica y Electrónica utilizando Frameworks de PHP.* [en línea]. Tesis Pregrado. Latacunga: Escuela Politécnica del Ejercito Extensión Latacunga. Departamento de Eléctrica y Electrónica. Escuela de Tecnología en Computación, agosto del 2013, pp.25. [Consulta 09 de 03 de 2017]. Disponible en <http://repositorio.espe.edu.ec/handle/21000/6768>.
11. **DÍAZ, Francisco Javier; QUEIRUGA, Claudia Alejandra; FAVA, Laura Andrea.** “Struts y JavaServer Faces, cara a cara” [en línea]. *En IX Workshop de Investigadores en Ciencias de la Computación.* 2007, p.3. [Consulta 01 de 02 de 2017]. Disponible en <http://sedici.unlp.edu.ar/handle/10915/20451>.
12. **ETCHART, Graciela, et al.** “Sistemas de reconocimiento biométricos, importancia del uso de estándares en entes estatales”. *En XIII Workshop de Investigadores en Ciencias de la Computación* [en línea]. 2011, pp.1-2. [Consulta 18 de 01 de 2017]. Disponible en <http://sedici.unlp.edu.ar/handle/10915/20052>.
13. **GALLEGO, Manuel Trigas.** *Metodología Scrum. Gestión de Proyectos Informáticos* [en línea]. 2012, pp.37-39,40-41. [Consulta 16 de 01 de 2017]. Disponible en <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>.
14. **GIMENO, Juan Manuel; GONZÁLEZ, José Luis.** *Introducción a Netbeans* [en línea]. 2011, pp.2-3. [Consulta 18 de 04 de 2017]. Disponible en <http://ocw.udl.cat/engineyria-i-arquitectura/engineyria-i-arquitectura/programacio-2/continguts-1/1-introduccioi81n-a-netbeans.pdf>.
15. **GUANOLUISA CARRERA, Jenny Germania.** *Análisis comparativo de la productividad entre los patrones de diseño Modelo Vista Controlador (MVC) y Modelo Vista Presentador (MVP) aplicado al desarrollo del Sistema Nómina de Empleados y Rol de Pagos de la Distribuidora Soria CA* [en línea]. Tesis de Pregrado. Riobamba: Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería en Sistemas, 3 de diciembre del 2014, pp.39-42. [Consulta 15 de 04 de 2017]. Disponible en <http://dspace.espech.edu.ec/handle/123456789/3583>.
16. **GUTIÉRREZ, Javier J.** *¿Qué es un framework Web?* [en línea]. [Consulta 05 de 03 de 2017]. Disponible en http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf.
17. **JIMÉNEZ, Luis R.; BAEZ, Evelio; PUIG, Osmany.** “Software para la identificación biométrica de los privados de libertad” [en línea]. *Universidad&Ciencia*, 2017, vol. 3, no 2, p.9.

[Consulta 28 de 04 de 2017]. Disponible en <http://revistas.unica.cu/index.php/uciencia/article/view/337>.

18. **KASIÁN, Fernando; REYES, Nora Susana.** Búsquedas por similitud en PostgreSQL [en línea]. En *XVIII Congreso Argentino de Ciencias de la Computación*. 2012, pp.4-5. [Consulta 27 de 04 de 2017]. Disponible en <http://dspace.ucacue.edu.ec/bitstream/reducacue/5443/4/Administraci%C3%B3n%20de%20la%20calidad%20en%20la%20fabriaci%C3%B3n%20de%20ballestas.pdf>.
19. **LA ADELACEN,** *Administración de la calidad en la fábrica de ballestas* [en línea]. Monografía Pregrado. Cuenca: Universidad Católica de Cuenca. Unidad Académica de Ingeniería. Química, biofarmacia, Industrias y Producción. Facultad de Ingeniería Industrial. 2013. p. IV. [Consulta 03 de 04 de 2017]. Disponible en <http://dspace.ucacue.edu.ec/bitstream/reducacue/5443/4/Administraci%C3%B3n%20de%20la%20calidad%20en%20la%20fabriaci%C3%B3n%20de%20ballestas.pdf>.
20. **MACIAS, Maria Veronica; REINOSO ESPINOZA, Jorge Arturo y otros.** Análisis de los modelos de calidad de software existentes y su apoyo al cumplimiento de requerimientos en empresas no dedicadas al desarrollo de software [en línea]. Artículo de Tesis de Pregrado. Guayaquil: Escuela Superior Politécnica del Litoral. Facultad de Ingeniería en Electricidad y Computación. 25 de febrero del 2009. [Consulta 01 de 04 de 2017]. Disponible en <http://www.dspace.espol.edu.ec/xmlui/handle/123456789/909>.
21. **MANCHADO, D. Serra; PUNTES, Daniel Franco.** *Estudio del servidor de aplicaciones Glassfish y de las aplicaciones J2EE* [en línea]. 2010, pp.96-97. [Consulta 06 de 04 de 2017]. Disponible en <https://core.ac.uk/download/pdf/13325903.pdf>.
22. **MELÉNDEZ REINA, Diego Marcelo; SÁNCHEZ TAPIA, Evelyn Valeria.** *Análisis, diseño y desarrollo de un sistema de administración control y monitoreo de equipos de Autenticación Biométrica* [en línea]. 2012. Tesis de Pregrado. Quito: Universidad Politécnica Salesiana. Escuela de Ingeniería de Sistemas Mención Telemática. febrero del 2012. pp. 4,14,18,78. [Consulta 11 de 03 de 2017]. Disponible en <http://dspace.ups.edu.ec/handle/123456789/1568>.
23. **MENA MENDOZA, Gonzalo.** *ISO 9126-3 Métricas Internas de la Calidad del Producto de Software* [en línea]. 2006. [Consulta 23 de 01 de 2017], pp.12-13. Disponible en http://mena.com.mx/gonzalo/maestria/calidad/presenta/iso_9126-3/.
24. **NTP-ISO IEC 9126-2.** *Norma Técnica peruana Calidad del producto - Métricas Externas*
25. **NTP-ISO IEC 9126-3.** *Norma Técnica peruana Calidad del producto - Métricas Interna*
26. **ORACLE CORPORATION AND/OR ITS AFFILIATES.** *Netbeans* [en línea]. [Consulta 07 de 02 de 2017]. Disponible en https://netbeans.org/index_es.html.

27. **PECH-MAY, Fernando; GOMEZ RODRIGUEZ, Mario A., y otros.** *Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces* [en línea]. Tabasco: Instituto Tecnológico Superior de los Ríos, 2010. pp. 9. [Consulta 15 de 04 de 2017]. Disponible en <http://www.tamps.cinvestav.mx/~fpech/sd/files/paper001.pdf>.
28. **PEREZ, Carlos A. Canepa, et al.** “Creación e implementación del módulo para la generación del modelo del negocio en la herramienta Case StarUML usando tecnologías de libre disponibilidad” [en línea]. *Revista de investigación de Sistemas e Informática*, vol. 9, no 1(2012), p. 57-67. [Consulta 07 de 01 de 2017]. Disponible en <http://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/5616>.
29. **REINOSO, Jorge; VIVANCO, Jorge; COBA, Carlos.** *Análisis de los modelos de calidad de software existentes y su apoyo al cumplimiento de requerimientos en empresas no dedicadas al desarrollo de software* [en línea]. Tesis de Pregrado. Guayaquil: Escuela Superior Politécnica del Litoral. Facultad de Ingeniería en Electricidad y Computación. Escuela de Ingeniería en computación especialización sistemas de información. 21 de mayo del 2010, p.19. [Consulta 20 de 02 de 2017]. Disponible en <http://www.dspace.espol.edu.ec/xmlui/handle/123456789/10151>.
30. **RUIZ, Gustavo Alberto, et al.** “Modelo de Evaluación de Calidad de Software Basado en Lógica Difusa, Aplicada a Métricas de Usabilidad de Acuerdo con la Norma ISO/IEC 9126” [en línea]. *RASI*, vol. 3, no 2(2006), pp. 25-29. [Consulta 23 de 02 de 2017]. Disponible en <https://pdfs.semanticscholar.org/ff04/48a237d0bb2fde289a53e77426b4cbf159ca.pdf>.
31. **SAP POWER DESIGNER.** *Base de datos y gestión de datos* [en línea]. [Consulta 11 de 03 de 2017]. Disponible en <https://www.sap.com/product/data-mgmt/powerdesigner-data-modeling-tools.product-capabilities.html>.
32. **TAPIA FLORES, Jonathan Gonzalo & CARRILLO FLORES, René Alfonso.** *Análisis del Framework Primefaces y la generación de código convencional JSF* [en línea]. 2016. Tesis de Pregrado. Quito: Universidad Central del Ecuador. Facultad de Informática y Electrónica. Escuela de Ingeniería en Informática, 18 de mayo del 2016. pp.4-6,101. [Consulta 12 de 03 de 2017]. Disponible en <http://www.dspace.uce.edu.ec/handle/25000/6074>.
33. **VALDIVIEZO SERRANO, Patricia Alexandra & GUACHO MINTA, María Alejandra.** *Análisis Comparativo de Tecnologías de Aplicaciones Web en el Entorno JSF y ADF* [en línea]. Caso Práctico IESS. Tesis de Pregrado. Riobamba: Escuela Superior Politécnica de Chimborazo. Facultad de Informática y Electrónica. Escuela de Ingeniería en Sistemas. 03 de febrero del 2012. p.36. [Consulta 06 de 01 de 2017]. Disponible en <http://dspace.esPOCH.edu.ec/handle/123456789/1490>.

34. **VASSDIGITAL.** *Definición de Power Designer* [en línea], 20 de 09 de 2012. [Consulta 06 de 03 de 2017.]. Disponible en <https://www.sap.com/product/data-mgmt/powerdesigner-data-modeling-tools.product-capabilities.html>.
35. **VIGO ROJAS, David A.; ACHING SAMATELO, Jorge L.** “Algoritmos para el reconocimiento de imágenes de huellas dactilares” [en línea]. *Revistas de Investigación UNMSM*, 2003, (Perú) vol. 12, pp 11-20. [Consulta 06 de 03 de 2017]. Disponible en <http://revistasinvestigacion.unmsm.edu.pe/index.php/electron/article/view/4008/31881>.
36. **VILLAVICENCIO CABEZAS, Mónica; ORTEGA CABRERA, Erick Enrique.** *Medición de la calidad de productos de software en un ambiente académico usando la norma iso/iec 9126* [en línea]. Artículo de Tesis de pregrado. Guayaquil: Escuela Superior Politécnica del Litoral. Facultad de Ingeniería en Electricidad y Computación. Escuela de Ingeniería en computación especialización sistemas de información. 27 de junio del 2011. pp.2-4. [Consulta 12 de 03 de 2017]. Disponible en <http://www.dspace.espol.edu.ec/xmlui/handle/123456789/16366>.
37. **VIVANCO VILLAMAR, Andrés Alejandro.** *Evaluación de calidad del sistema Integrado para casas de valores SICAV de la bolsa de valores de Quito utilizando la norma ISO/IEC 14598* [en línea], 2011. Tesis de Pregrado. Quito: Escuela Politécnica Nacional. Facultad de Ingeniería. Escuela de Ingeniería en Sistemas Informáticos y de Computación. Agosto del 2011, pp.1-3,20,30-31,40-41,123-124,175,181. [Consulta 08 de 03 de 2017]. Disponible en <http://bibdigital.epn.edu.ec/handle/15000/4329>.

ANEXOS

Anexo A. Plan de Actividades

Id	Mod de tarea	Nombre de tarea	Duración	Comienzo	Fin	Gantt Chart Timeline											
						abril 14/03	09/05	julio 04/07	29/08	octubre 24/10	enero 19/12	13/02	abril 10/04	julio 05/06	31/07		
1	➤	INICIO DEL PROYECTO	110 días	jue 19/01/17	mié 21/06/17												
2	➤	Sprint 0	10 días	jue 19/01/17	mié 01/02/17												
3	➤	Estudio de la tecnología a utilizar	2 días	jue 19/01/17	vie 20/01/17												
4	➤	Análisis y Gestión de Riesgos	4 días	lun 23/01/17	jue 26/01/17												
5	➤	Diseño de la Base de datos	4 días	vie 27/01/17	mié 01/02/17												
6	➤	Sprint 1	10 días	jue 02/02/17	mié 15/02/17												
7	➤	Diseño de la Arquitectura del Sistema	4 días	jue 02/02/17	mar 07/02/17												
8	➤	Definición del Estándar de Codificación	2 días	mié 08/02/17	jue 09/02/17												
9	➤	Diseño del diagrama de Clases	2 días	vie 10/02/17	lun 13/02/17												
10	➤	Diseño de Interfaces	2 días	mar 14/02/17	mié 15/02/17												
11	➤	Sprint 2	10 días	jue 16/02/17	mié 01/03/17												
12	➤	Ingreso datos empleado	4 días	jue 16/02/17	mar 21/02/17												
13	➤	Actualización datos empleado	2 días	mié 22/02/17	jue 23/02/17												
14	➤	Listar datos empleados	2 días	vie 24/02/17	lun 27/02/17												
15	➤	Eliminación datos empleado	2 días	mar 28/02/17	mié 01/03/17												
16	➤	Sprint 3	10 días	jue 02/03/17	mié 15/03/17												
17	➤	Reingreso de datos personales	2 días	jue 02/03/17	vie 03/03/17												
18	➤	Actualizar el departamento de empleados de forma masiva	3 días	lun 06/03/17	mié 08/03/17												
19	➤	Generar lista de cumpleaños del mes	3 días	jue 09/03/17	lun 13/03/17												
20	➤	Enviar notificación de felicitación de cumpleaños a cada email	2 días	mar 14/03/17	mié 15/03/17												

Proyecto: planificacion-tentativ Fecha: vie 31/03/17	Tarea		Resumen inactivo		Tareas externas	
	División		Tarea manual		Hito externo	
	Hito		solo duración		Fecha límite	
	Resumen		Informe de resumen manual		Progreso	
	Resumen del proyecto		Resumen manual		Progreso manual	
	Tarea inactiva		solo el comienzo			
	Hito inactivo		solo fin			

ANEXO B: Diccionario de Datos

Tabla 74: Diccionario de datos de Tabla asistencia

table_	table_	table_	column_name	data_type	character_	character_	numeric_
catalog	chema	name			maximum	octet	precision
					_length	_length	
FUNAMI	fundacion	asistencia	id_asistencia	integer			32
FUNAMI	fundacion	asistencia	id_historial_periodo_empleado	integer			32
FUNAMI	fundacion	asistencia	id_dia_registro	integer			32
FUNAMI	fundacion	asistencia	estado_asistencia	character varying	16	64	
FUNAMI	fundacion	asistencia	estado_asistencia_hora	character varying	16	64	
FUNAMI	fundacion	asistencia	hora_entrada	time without time zone			
FUNAMI	fundacion	asistencia	hora_almuerzo_salida	time without time zone			
FUNAMI	fundacion	asistencia	hora_almuerzo_regreso	time without time zone			
FUNAMI	fundacion	asistencia	hora_salida	time without time zone			
FUNAMI	fundacion	asistencia	es_activo	boolean			
FUNAMI	fundacion	asistencia	fecha_creacion	date			
FUNAMI	fundacion	asistencia	id_usuario_creacion	integer			32
FUNAMI	fundacion	asistencia	id_subempresa_creacion	integer			32
FUNAMI	fundacion	asistencia	fecha_actualizacion	date			
FUNAMI	fundacion	asistencia	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	asistencia	id_subempresa_actualizacion	integer			32
FUNAMI	fundacion	asistencia	justificacion_estado	character varying	256	1024	

Realizado por: Erika R. 2017

Tabla 75: Diccionario de datos de Tabla atraso

table_	table_	table_	column_	data_type	character_	Carácte	numeric
catalog	schema	name	name		maximum	r	_
					length	_octet	precision
						_length	
FUNAMI	fundacion	atraso	id_atraso	integer			32
FUNAMI	fundacion	atraso	id_historial_periodo_empleado	integer			32
FUNAMI	fundacion	atraso	fecha_registro	date			
FUNAMI	fundacion	atraso	hora_registro	time without time zone			
FUNAMI	fundacion	atraso	duracion_min	integer			32
FUNAMI	fundacion	atraso	estado_atraso	character varying	16	64	
FUNAMI	fundacion	atraso	es_activo	boolean			
FUNAMI	fundacion	atraso	fecha_creacion	date			
FUNAMI	fundacion	atraso	id_usuario_creacion	integer			32
FUNAMI	fundacion	atraso	id_subempresa_creacion	integer			32
FUNAMI	fundacion	atraso	fecha_actualizacion	date			
FUNAMI	fundacion	atraso	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	atraso	id_subempresa_actualizacion	integer			32
FUNAMI	fundacion	atraso	justificacion_estado	character varying	256	1024	

Realizado por: Erika R. 2017

Tabla 76: Diccionario de datos de Tabla biometria

table_catalog	table_schema	table_name	column_name	data_type	numeric_precision
FUNAMI	fundacion	biometria	id_biometria	integer	32
FUNAMI	fundacion	biometria	id_empleado	integer	32
FUNAMI	fundacion	biometria	image_pulgar_derecho	bytea	
FUNAMI	fundacion	biometria	image_pulgar_izquierdo	bytea	
FUNAMI	fundacion	biometria	image_indice_derecho	bytea	
FUNAMI	fundacion	biometria	image_indice_izquierdo	bytea	
FUNAMI	fundacion	biometria	es_activo	boolean	
FUNAMI	fundacion	biometria	fecha_creacion	date	
FUNAMI	fundacion	biometria	id_usuario_creacion	integer	32
FUNAMI	fundacion	biometria	id_subempresa_creacion	integer	32
FUNAMI	fundacion	biometria	fecha_actualizacion	date	
FUNAMI	fundacion	biometria	id_usuario_actualizacion	integer	32
FUNAMI	fundacion	biometria	id_subempresa_actualizacion	integer	32

Realizado por: Erika R. 2017

Tabla 77: Diccionario de datos de Tabla departamento

table_catalog	table_schema	table_name	column_name	data_type	Carácter_Máximo_length	Carácter_octet_length	numeric_precision
FUNAMI	fundacion	departamento	id_departamento	integer			32
FUNAMI	fundacion	departamento	id_subempresa	integer			32
FUNAMI	fundacion	departamento	nombre	character varying	64	256	
FUNAMI	fundacion	departamento	descripcion	character varying	128	512	
FUNAMI	fundacion	departamento	responsable	character varying	64	256	
FUNAMI	fundacion	departamento	id_departamento	integer			32
FUNAMI	fundacion	departamento	id_subempresa	integer			32
FUNAMI	fundacion	departamento	nombre	character varying	64	256	
FUNAMI	fundacion	departamento	descripcion	character varying	128	512	
FUNAMI	fundacion	departamento	responsable	character varying	64	256	

Realizado por: Erika R. 2017

Tabla 78: Diccionario de datos de Tabla dia_registro

table_ catalog	table_ schema	table_name	column_ name	data_ type	Character _maximum _length	Character _octet _length	Numeric _precision
FUNAMI	fundacion	dia_registro	id_dia_registro	integer			32
FUNAMI	fundacion	dia_registro	id_periodo	integer			32
FUNAMI	fundacion	dia_registro	fecha_registro	date			
FUNAMI	fundacion	dia_registro	descripcion	character varying	128	512	
FUNAMI	fundacion	dia_registro	es_activo	boolean			
FUNAMI	fundacion	dia_registro	fecha_creacion	date			
FUNAMI	fundacion	dia_registro	id_usuario_creacion	integer			32
FUNAMI	fundacion	dia_registro	id_subempresa_creacion	integer			32
FUNAMI	fundacion	dia_registro	fecha_actualizacion	date			
FUNAMI	fundacion	dia_registro	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	dia_registro	id_subempresa_actualizacion	integer			32

Realizado por: Erika R. 2017

Tabla 79: Diccionario de datos de Tabla dirección_empleado

table_	table_	table_name	column_name	data_type	character_	character_	numeric_
catalog	schema				maximum_	octet_	precision
					length	length	
FUNAMI	fundacion	direccion_empleado	id_direccion_e mpleado	integer			32
FUNAMI	fundacion	direccion_empleado	id_empleado	integer			32
FUNAMI	fundacion	direccion_empleado	codigo_provinc ia	character varying	16	64	
FUNAMI	fundacion	direccion_empleado	codigo_canton	character varying	16	64	
FUNAMI	fundacion	direccion_empleado	tipo_direccion	character varying	16	64	
FUNAMI	fundacion	direccion_empleado	calle	character varying	64	256	
FUNAMI	fundacion	direccion_empleado	numero	character varying	32	128	
FUNAMI	fundacion	direccion_empleado	es_activo	boolean			
FUNAMI	fundacion	direccion_empleado	fecha_creacion	date			
FUNAMI	fundacion	direccion_empleado	id_usuario_crea cion	integer			32
FUNAMI	fundacion	direccion_empleado	id_subempresa_ creacion	integer			32
FUNAMI	fundacion	direccion_empleado	fecha_actualiza cion	date			
FUNAMI	fundacion	direccion_empleado	id_usuario_actu alizacion	integer			32
FUNAMI	fundacion	direccion_empleado	id_subempresa_ actualizacion	integer			32

Realizado por: Erika R. 2017

Tabla 80: Diccionario de datos de Tabla discapacidad_empleado

table_ catalog	table_ schema	table_ name	column_ name	data_ type	character_ maximum_ length	character_ octet_ length	numeric_ precision
FUNAMI	fundacion	discapacidad_empleado	id_discapacidad_empleado	integer			32
FUNAMI	fundacion	discapacidad_empleado	id_empleado	integer			32
FUNAMI	fundacion	discapacidad_empleado	tipo	character varying	16	64	
FUNAMI	fundacion	discapacidad_empleado	porcentaje	numeric			19
FUNAMI	fundacion	discapacidad_empleado	identificacion_responsable	character varying	16	64	
FUNAMI	fundacion	discapacidad_empleado	es_activo	boolean			
FUNAMI	fundacion	discapacidad_empleado	fecha_creacion	date			
FUNAMI	fundacion	discapacidad_empleado	id_usuario_creacion	integer			32
FUNAMI	fundacion	discapacidad_empleado	id_subempresa_creacion	integer			32
FUNAMI	fundacion	discapacidad_empleado	fecha_actualizacion	date			
FUNAMI	fundacion	discapacidad_empleado	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	discapacidad_empleado	id_subempresa_actualizacion	integer			32

Realizado por: Erika R. 2017

Tabla 81: Diccionario de datos de Tabla dispositivo

table_catalog	table_schema	table_name	column_name	data_type	character_maximum_length	character_octet_length	numeric_precision
FUNAMI	fundacion	dispositivo	id_dispositivo	integer			32
FUNAMI	fundacion	dispositivo	direccion_ip	character varying	32	128	
FUNAMI	fundacion	dispositivo	puerto	character varying	32	128	
FUNAMI	fundacion	dispositivo	es_activo	boolean			
FUNAMI	fundacion	dispositivo	fecha_creacion	date			
FUNAMI	fundacion	dispositivo	id_usuario_creacion	integer			32
FUNAMI	fundacion	dispositivo	id_subempresa_creacion	integer			32
FUNAMI	fundacion	dispositivo	fecha_actualizacion	date			
FUNAMI	fundacion	dispositivo	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	dispositivo	id_subempresa_actualizacion	integer			32

Realizado por: Erika R. 2017

Tabla 82: Diccionario de datos de Tabla dispositivo_ubicación

table_catalog	table_schema	table_name	column_name	data_type	character_maximum_length	character_octet_length	numeric_precision
FUNAMI	fundacion	dispositivo_ubicacion	id_dispositivo_ubicacion	integer			32
FUNAMI	fundacion	dispositivo_ubicacion	id_empleado	integer			32
FUNAMI	fundacion	dispositivo_ubicacion	tipo	character_varying	16	64	
FUNAMI	fundacion	dispositivo_ubicacion	valor	character_varying	64	256	
FUNAMI	fundacion	dispositivo_ubicacion	es_activo	boolean			
FUNAMI	fundacion	dispositivo_ubicacion	fecha_creacion	date			
FUNAMI	fundacion	dispositivo_ubicacion	id_usuario_creacion	integer			32
FUNAMI	fundacion	dispositivo_ubicacion	id_subempresa_creacion	integer			32
FUNAMI	fundacion	dispositivo_ubicacion	fecha_actualizacion	date			
FUNAMI	fundacion	dispositivo_ubicacion	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	dispositivo_ubicacion	id_subempresa_actualizacion	integer			32

Realizado por: Erika R. 2017

Tabla 83: Diccionario de datos de Tabla empleado

table_catalog	table_schema	table_name	column_name	data_type	character_maximum_length	character_octet_length	numeric_precision
FUNAMI	fundacion	empleado	id_empleado	integer			32
FUNAMI	fundacion	empleado	identificacion	character_varying	16	64	
FUNAMI	fundacion	empleado	primer_nombre	character_varying	32	128	
FUNAMI	fundacion	empleado	segundo_nombre	character_varying	32	128	
FUNAMI	fundacion	empleado	apellido_paterno	character_varying	32	128	
FUNAMI	fundacion	empleado	apellido_materno	character_varying	32	128	
FUNAMI	fundacion	empleado	genero	character_varying	16	64	
FUNAMI	fundacion	empleado	estado_civil	character_varying	16	64	
FUNAMI	fundacion	empleado	cargas_familiares	integer			32
FUNAMI	fundacion	empleado	fecha_nacimiento	date			
FUNAMI	fundacion	empleado	pais	character_varying	32	128	
FUNAMI	fundacion	empleado	es_activo	boolean			
FUNAMI	fundacion	empleado	comvenio_doble	boolean			
FUNAMI	fundacion	empleado	fecha_creacion	date			

FUNAMI	fundacion	empleado	id_subempresa_actual	integer			32
FUNAMI	fundacion	empleado	id_usuario_creacion	integer			32
FUNAMI	fundacion	empleado	id_subempresa_creacion	integer			32
FUNAMI	fundacion	empleado	fecha_actualizacion	date			
FUNAMI	fundacion	empleado	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	empleado	id_subempresa_actualizacion	integer			32

Realizado por: Erika R. 2017

Tabla 84: Diccionario de datos de Tabla empleado_horario_laborable

table_	table_	table_name	column_name	data_type	numeric_
catalog	schema				precision
FUNAMI	fundacion	empleado_horario_laborable	id_empleado_horario_laborable	integer	32
FUNAMI	fundacion	empleado_horario_laborable	id_empleado	integer	32
FUNAMI	fundacion	empleado_horario_laborable	id_horario_laborable	integer	32
FUNAMI	fundacion	empleado_horario_laborable	es_activo	boolean	

Realizado por: Erika R. 2017

Tabla 85: Diccionario de datos de Tabla historial laboral

table_catalog	table_schema	table_name	column_name	data_type	numeric_precision
FUNAMI	fundacion	historial_laboral	id_historial_laboral	integer	32
FUNAMI	fundacion	historial_laboral	id_empleado	integer	32
FUNAMI	fundacion	historial_laboral	total_num_dias_vacacion	numeric	19
FUNAMI	fundacion	historial_laboral	numero_asistencia	integer	32
FUNAMI	fundacion	historial_laboral	numero_faltas	integer	32
FUNAMI	fundacion	historial_laboral	numero_permiso	integer	32
FUNAMI	fundacion	historial_laboral	numero_atrasos	integer	32
FUNAMI	fundacion	historial_laboral	es_activo	boolean	
FUNAMI	fundacion	historial_laboral	fecha_creacion	date	
FUNAMI	fundacion	historial_laboral	id_usuario_creacion	integer	32
FUNAMI	fundacion	historial_laboral	id_subempresa_creacion	integer	32
FUNAMI	fundacion	historial_laboral	fecha_actualizacion	date	
FUNAMI	fundacion	historial_laboral	id_usuario_actualizacion	integer	32
FUNAMI	fundacion	historial_laboral	id_subempresa_actualizacion	integer	32

Realizado por: Erika R. 2017

Tabla 86: Diccionario de datos de Tabla historial_periodo_empleado

table_catalog	table_schema	table_name	column_name	data_type	numeric_precision
FUNAMI	fundacion	historial_periodo_empleado	id_historial_periodo_empleado	integer	32
FUNAMI	fundacion	historial_periodo_empleado	id_historial_laboral	integer	32
FUNAMI	fundacion	historial_periodo_empleado	id_periodo	integer	32
FUNAMI	fundacion	historial_periodo_empleado	num_dias_vacacion	numeric	10
FUNAMI	fundacion	historial_periodo_empleado	num_dias_asistencia	integer	32
FUNAMI	fundacion	historial_periodo_empleado	num_dias_falta	integer	32
FUNAMI	fundacion	historial_periodo_empleado	num_permisos	integer	32
FUNAMI	fundacion	historial_periodo_empleado	numero_atrasos	integer	32
FUNAMI	fundacion	historial_periodo_empleado	es_activo	boolean	
FUNAMI	fundacion	historial_periodo_empleado	fecha_creacion	date	
FUNAMI	fundacion	historial_periodo_empleado	id_usuario_creacion	integer	32
FUNAMI	fundacion	historial_periodo_empleado	id_subempresa_creacion	integer	32
FUNAMI	fundacion	historial_periodo_empleado	fecha_actualizacion	date	
FUNAMI	fundacion	historial_periodo_empleado	id_usuario_actualizacion	integer	32
FUNAMI	fundacion	historial_periodo_empleado	id_subempresa_actualizacion	integer	32

Realizado por: Erika R. 2017

Tabla 87: Diccionario de datos de Tabla horario_laborable

table_catalog	table_schema	table_name	column_name	data_type	character_maximum_length	character_octet_length	numeric_precision
FUNAMI	fundacion	horario_laborable	id_horario_laborable	integer			32
FUNAMI	fundacion	horario_laborable	nombre	character varying	64	256	
FUNAMI	fundacion	horario_laborable	descripcion	character varying	256	1024	
FUNAMI	fundacion	horario_laborable	hora_entrada	time without time zone			
FUNAMI	fundacion	horario_laborable	hora_almuerzo_salida	time without time zone			
FUNAMI	fundacion	horario_laborable	hora_almuerzo_regreso	time without time zone			
FUNAMI	fundacion	horario_laborable	hora_salida	time without time zone			
FUNAMI	fundacion	horario_laborable	es_activo	boolean			
FUNAMI	fundacion	horario_laborable	numero_horas_laboral	integer			32
FUNAMI	fundacion	horario_laborable	fecha_creacion	date			
FUNAMI	fundacion	horario_laborable	id_usuario_cre	integer			32

			acion			
FUNAMI	fundacion	horario_laborable	id_subempresa_creacion	integer		32
FUNAMI	fundacion	horario_laborable	fecha_actualizacion	date		
FUNAMI	fundacion	horario_laborable	id_usuario_actualizacion	integer		32
FUNAMI	fundacion	horario_laborable	id_subempresa_actualizacion	integer		32

Realizado por: Erika R. 2017

Tabla 88: Diccionario de datos de Tabla periodo

table_catalog	table_schema	table_name	column_name	data_type	numeric_precision
FUNAMI	fundacion	periodo	id_periodo	integer	32
FUNAMI	fundacion	periodo	fecha_inicio	date	
FUNAMI	fundacion	periodo	fecha_fin	date	
FUNAMI	fundacion	periodo	es_activo	boolean	
FUNAMI	fundacion	periodo	es_vigente	boolean	
FUNAMI	fundacion	periodo	numero_dias_trabajar	integer	32
FUNAMI	fundacion	periodo	fecha_creacion	date	
FUNAMI	fundacion	periodo	id_usuario_creacion	integer	32
FUNAMI	fundacion	periodo	id_subempresa_creacion	integer	32
FUNAMI	fundacion	periodo	fecha_actualizacion	date	
FUNAMI	fundacion	periodo	id_usuario_actualizacion	integer	32
FUNAMI	fundacion	periodo	id_subempresa_actualizacion	integer	32

Realizado por: Erika R. 2017

Tabla 89: Diccionario de datos de Tabla permiso

table_catalog	table_schema	table_name	column_name	data_type	character_maximum_length	character_octet_length	numeric_precision
FUNAMI	fundacion	permiso	id_permiso	integer			32
FUNAMI	fundacion	permiso	id_historial_periodo_empleado	integer			32
FUNAMI	fundacion	permiso	id_dia_registro	integer			32
FUNAMI	fundacion	permiso	justificacion	character_varying	256	1024	
FUNAMI	fundacion	permiso	fecha_permiso	date			
FUNAMI	fundacion	permiso	estado_permiso	character_varying	16	64	
FUNAMI	fundacion	permiso	es_activo	boolean			
FUNAMI	fundacion	permiso	fecha_creacion	date			
FUNAMI	fundacion	permiso	id_usuario_creacion	integer			32
FUNAMI	fundacion	permiso	id_subempresa_creacion	integer			32
FUNAMI	fundacion	permiso	fecha_actualizacion	date			
FUNAMI	fundacion	permiso	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	permiso	id_subempresa_actualizacion	integer			32

FUNAMI	fundacion	permiso	justificacion_estado	character varying	256	1024	
FUNAMI	fundacion	permiso	tipo_permiso	character varying	16	64	

Realizado por: Erika R. 2017

Tabla 90: Diccionario de datos de Tabla registro

table_catalog	table_schema	table_name	column_name	data_type	character_maximum_length	character_octet_length	numeric_precision
FUNAMI	fundacion	dia_registro	id_dia_registro	integer			32
FUNAMI	fundacion	dia_registro	id_periodo	integer			32
FUNAMI	fundacion	dia_registro	fecha_registro	date			
FUNAMI	fundacion	dia_registro	descripcion	character varying	128	512	
FUNAMI	fundacion	dia_registro	es_activo	boolean			
FUNAMI	fundacion	dia_registro	fecha_creacion	date			
FUNAMI	fundacion	dia_registro	id_usuario_creacion	integer			32
FUNAMI	fundacion	dia_registro	id_subempresa_creacion	integer			32
FUNAMI	fundacion	dia_registro	fecha_actualizacion	date			
FUNAMI	fundacion	dia_registro	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	dia_registro	id_subempresa_actualizacion	integer			32

Realizado por: Erika R. 2017

Tabla 91: Diccionario de datos de Tabla solicitud_vacación

table_catalog	table_schema	table_name	column_name	data_type	character_maximum_length	character_octet_length	numeric_precision
FUNAMI	fundacion	solicitud_vacacion	id_solicitud_vacacion	integer			32
FUNAMI	fundacion	solicitud_vacacion	id_historial_laboral	integer			32
FUNAMI	fundacion	solicitud_vacacion	id_empleado	integer			32
FUNAMI	fundacion	solicitud_vacacion	fecha_salida	date			
FUNAMI	fundacion	solicitud_vacacion	fecha_retorno	date			
FUNAMI	fundacion	solicitud_vacacion	numero_dias	integer			32
FUNAMI	fundacion	solicitud_vacacion	estado_solicitud	character varying	16	64	
FUNAMI	fundacion	solicitud_vacacion	autorizacion_jefe_inmediato	boolean			
FUNAMI	fundacion	solicitud_vacacion	fecha_autorizacion_jefe_inmediato	date			
FUNAMI	fundacion	solicitud_vacacion	autorizacion_departamento	boolean			
FUNAMI	fundacion	solicitud_vacacion	fecha_autorizacion_departamento	date			
FUNAMI	fundacion	solicitud_vacacion	autorizacion_rrhh	boolean			
FUNAMI	fundacion	solicitud_vacacion	es_activo	boolean			

FUNAMI	fundacion	solicitud_vacacion	fecha_autorizacion_rrhh	date			
FUNAMI	fundacion	solicitud_vacacion	fecha_creacion	date			
FUNAMI	fundacion	solicitud_vacacion	id_usuario_creacion	integer			32
FUNAMI	fundacion	solicitud_vacacion	id_subempresa_creacion	integer			32
FUNAMI	fundacion	solicitud_vacacion	fecha_actualizacion	date			
FUNAMI	fundacion	solicitud_vacacion	id_usuario_actualizacion	integer			32
FUNAMI	fundacion	solicitud_vacacion	id_subempresa_actualizacion	integer			32

Realizado por: Erika R. 2017

ANEXO C: Gestión de Riesgos

Tabla 92-3: Gestión de Riesgo 1

HOJA DE GESTIÓN DEL RIESGO			
ID. DEL RIESGO: R1		FECHA: 2016/05/26	
Probabilidad: Baja Valor: 1	Impacto: Crítico Valor: 4	Exposición: Media Valor: 4	Prioridad: 1
DESCRIPCIÓN: Cancelación de actividades del negocio.			
REFINAMIENTO:			
Causas:			
<ul style="list-style-type: none"> El personal administrativo del negocio no asiste a las reuniones con el equipo de desarrollo. No se brinda toda la información necesaria para realizar el proyecto. 			
Consecuencias:			
<ul style="list-style-type: none"> Fracaso del Proyecto. 			
REDUCCIÓN:			
<ul style="list-style-type: none"> Planificar con anterioridad las fechas en La cuales el personal administrativo puedan asistir a las reuniones. 			
SUPERVISION:			
<ul style="list-style-type: none"> Verificar el cumplimiento de la planificación de actividades. Verificar el cumplimiento de la asistencia del personal administrativo. Comprobar que en el desarrollo los costos no se salgan del presupuesto. 			
GESTIÓN:			
<ul style="list-style-type: none"> Reuniones y socialización con el equipo y con el personal administrativo. Renegociación oportuna de costos. 			
ESTADO ACTUAL:			
	Fase de reducción iniciada	<input checked="" type="checkbox"/>	
	Fase de Supervisión iniciada	<input checked="" type="checkbox"/>	
	Gestionando el riesgo	<input type="checkbox"/>	
RESPONSABLE: Erika Reina			

Realizado por: Erika R. 2017

Tabla 93-3: Gestión de Riesgo 2

HOJA DE GESTIÓN DEL RIESGO			
ID. DEL RIESGO: R2		FECHA: 2016/05/26	
Probabilidad: Baja Valor: 1	Impacto: Alto Valor: 3	Exposición: Media Valor: 3	Prioridad: 2
DESCRIPCIÓN: Pérdida o robo de los equipos de trabajo.			
REFINAMIENTO:			
Causas:			
<ul style="list-style-type: none"> • Falta de seguridad en el lugar de desarrollo • Equipos sin identificación referente a la fundación • Mala contratación de 			
Consecuencias:			
<ul style="list-style-type: none"> • Pérdida de información • Pérdida económica • Retraso del proyecto 			
REDUCCIÓN:			
<ul style="list-style-type: none"> • Tener un sistema de seguridad eficiente para el lugar de desarrollo • Realizar copias de seguridad de toda la información la nube • Contratación de personal con diversas referencias 			
SUPERVISIÓN:			
<ul style="list-style-type: none"> • Mantener la seguridad los equipos mediante cámaras de vigilancia 			
GESTIÓN:			
<ul style="list-style-type: none"> • Recuperar la información perdida mediante las copias de seguridad • Adquirir nuevos equipos para continuar con el desarrollo del proyecto 			
ESTADO ACTUAL:			
Fase de reducción iniciada		<input checked="" type="checkbox"/>	
Fase de Supervisión iniciada		<input checked="" type="checkbox"/>	
Gestionando el riesgo		<input type="checkbox"/>	
RESPONSABLE: Erika Reina			

Tabla 94-3: Gestión de Riesgo 3

HOJA DE GESTIÓN DEL RIESGO			
ID. DEL RIESGO: R3		FECHA: 2016/05/26	
Probabilidad: Media Valor: 2	Impacto: Moderado Valor: 2	Exposición: Media Valor: 4	Prioridad: 3
DESCRIPCIÓN: Falla en los servidores.			
REFINAMIENTO:			
Causas:			
<ul style="list-style-type: none"> • Problemas con el servicio eléctrico. • Inconvenientes con los proveedores de servicios de internet. • Falta de mantenimiento adecuado a los servidores. 			
Consecuencias:			
<ul style="list-style-type: none"> • Perdidas económicas. • Perdida de la información. • Errores en las transacciones. 			
REDUCCIÓN:			
<ul style="list-style-type: none"> • Planificar con anterioridad las fechas en la cuales el personal administrativo puedan asistir a las reuniones. • Organizar un cronograma de mantenimiento en los servidores para asegurar la disponibilidad de los mismos. 			
SUPERVISIÓN:			
<ul style="list-style-type: none"> • Verificar el cumplimiento de la planificación de actividades. • Verificar el cumplimiento de la asistencia del personal administrativo. • Verificar los problemas técnicos con los servidores. 			
GESTIÓN:			
<ul style="list-style-type: none"> • Reuniones y socialización con el equipo y con el personal administrativo. • Recurrir a un nuevo proveedor de servicio de internet. • Contratar personal técnico para el mantenimiento de los servidores. 			
ESTADO ACTUAL:			
	Fase de reducción iniciada	<input checked="" type="checkbox"/>	
	Fase de Supervisión iniciada	<input checked="" type="checkbox"/>	
	Gestionando el riesgo	<input type="checkbox"/>	
RESPONSABLE: Erika Reina			

Realizado por: Erika R. 2017

Tabla 95-3: Gestión de Riesgo 4

HOJA DE GESTIÓN DEL RIESGO			
ID. DEL RIESGO: R4		FECHA: 2016/05/26	
Probabilidad: Baja Valor: 1	Impacto: Alto Valor: 3	Exposición: Media Valor: 3	Prioridad: 4
DESCRIPCIÓN: Cambio de políticas de la empresa que incurren en las metas y objetivos del proyecto.			
REFINAMIENTO:			
Causas:			
<ul style="list-style-type: none"> • Cuestiones internas de la Fundación. 			
Consecuencias:			
<ul style="list-style-type: none"> • Cambio en las funcionalidades ya desarrolladas del sistema. • Retraso de la ejecución del proyecto. • Incertidumbre no esperada durante el desarrollo. 			
REDUCCIÓN:			
<ul style="list-style-type: none"> • Desarrollo de un proyecto con calidad desde sus inicios para que sea modificable, interoperable e integrable en todas sus funcionalidades. 			
SUPERVISIÓN:			
<ul style="list-style-type: none"> • Verificar el desarrollo correcto de funcionalidades. • Realizar pruebas eficaces de exploración para cada funcionalidad. 			
GESTIÓN:			
<ul style="list-style-type: none"> • Reuniones y socialización con el equipo y con el Gerente de la Fundación. • Renegociación oportuna de costos debido a cambios de mayor alcance en los requerimientos. • Modificar el sistema adecuadamente sin poner en riesgo las funcionalidades. • Realizar pruebas constantes. 			
ESTADO ACTUAL:			
		Fase de reducción iniciada	<input checked="" type="checkbox"/>
		Fase de Supervisión iniciada	<input checked="" type="checkbox"/>
		Gestionando el riesgo	<input type="checkbox"/>
RESPONSABLES: Erika Reina			

Realizado por: Erika R. 2017

Tabla 96-3: Gestión de Riesgo 5

HOJA DE GESTIÓN DEL RIESGO			
ID. DEL RIESGO: R5		FECHA: 2016/05/26	
Probabilidad: Alta Valor: 3	Impacto: Alto Valor: 3	Exposición: Alta Valor: 9	Prioridad: 5
DESCRIPCIÓN: El usuario cambia constantemente los requerimientos propuestos.			
REFINAMIENTO:			
Causas:			
<ul style="list-style-type: none"> • Problemas con la comunicación entre el grupo de trabajo y el cliente. • Cambios en cuanto a la manera de llevar la información en la fundación. 			
Consecuencias:			
<ul style="list-style-type: none"> • Rediseño del Proyecto • Re-planificación del Proyecto. • Incremento de costos. • Retraso del proyecto 			
REDUCCIÓN:			
<ul style="list-style-type: none"> • Reuniones constantes con el cliente para así lograr una mejor comunicación y por ende un entendimiento claro de lo que se requiere. • Planificar el presupuesto de todo el proyecto para así cuando el cliente requiera cambiar algo se agregue ello al costo final del proyecto. 			
SUPERVISIÓN:			
<ul style="list-style-type: none"> • Verificar el cumplimiento correcto de todos los requerimientos propuestos por el cliente. • Verificar en el desarrollo que los costos no se salgan del presupuesto. 			
GESTIÓN:			
<ul style="list-style-type: none"> • Reuniones y socialización con el equipo y con el Gerente de la Fundación. • Renegociación oportuna de costos. • Amoldarse a la situación económica establecida. 			
ESTADO ACTUAL:			
		Fase de reducción iniciada	<input checked="" type="checkbox"/>
		Fase de Supervisión iniciada	<input checked="" type="checkbox"/>
		Gestionando el riesgo	<input type="checkbox"/>
RESPONSABLE: Erika Reina			

Realizado por: Erika R. 2017

ANEXO D: Historias de Usuario

Sprint 2

Historia de usuario 1

Tabla 97-3 HU-01 Ingreso de empleados

HISTORIA USUARIO	
Número: HU-01	Nombre Historia de Usuario: Ingresar la información de empleados.
Usuario: Administrador	Sprint Asignado: 02
Fecha Inicio: 16/02/17	Fecha Fin: 21/02/17
Descripción: Como administrador necesito ingresar la información de empleados, para poder tenerla almacenada y actualizada en la base de datos para de esa manera utilizarla cuando sea requerida.	
Pruebas de aceptación: <ul style="list-style-type: none">- Ingreso de empleados correctamente y re direccionar a la página correspondiente.- Introducir un empleado ya registrado y emitir el mensaje correspondiente.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-01 se define en la siguiente tabla.

Tabla 98-3 Tabla de Actividades para HU-01

FECHA	ACTIVIDAD	TIEMPO
16/02/17- 21/02/17	Tarea 1: Desarrollo del ingreso de empleados. <ul style="list-style-type: none">• Creación de la clase NegocioEmpleado y el método crear(), en ella.• Creación del método rederizarNuevoEmpleado() en el ControladorEmpleado• Creación del formularioEmpleado en la vista EmpleadoAdministración	32 horas

Realizado por: Erika R. 2017

Tabla 99-3 Tarea de Ingeniería 1 para la HU-01

TAREAS DE INGENIERÍA	
SPRINT: 02	Número de Tarea: 01
Nombre de Historia de Usuario: Ingresar la información de empleados.	
Nombre de la Tarea: Implementación del ingreso de Empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 16/02/17	Fecha Fin: 21/02/17
Descripción <ul style="list-style-type: none"> • Creación de la clase NegocioEmpleado y el método crear(), en ella. • Creación del método rederizarNuevoEmpleado() en el ControladorEmpleado • Creación del formularioEmpleado en la vista EmpleadoAdministración 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Introducir los datos correctos del empleado y redireccionar al listado de los empleados. • Introducir un empleado ya registrado y emitir el mensaje correspondiente 	

Realizado por: Erika R. 2017

Tabla 100-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación del ingreso de Empleados.
Nombre de la Prueba: Introducir los datos correctos del empleado y redireccionar al listado de los empleados.	
Responsable: Erika Reina	Fecha: 21/02/17
Descripción: Realizar el ingreso de los datos primordiales para la nómina de los empleados de la Fundación.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado • Clic en el botón Nuevo • Ingresar los Datos requeridos Identificación: 0804344661 Primer nombre: Erika Apellido Paterno: Reina Sexo: Femenino 	

<p>Estado Civil: Soltera Pais: Ecuador</p> <ul style="list-style-type: none"> • Clic en Grabar Todo/ Grabar
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Empleado Ingresado.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 101-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación del ingreso de Empleados.
Nombre de la Prueba: Introducir un empleado ya registrado y emitir el mensaje correspondiente.	
Responsable: Erika Reina	Fecha: 21/02/17
Descripción: Realizar el ingreso de un empleado que ya se encuentre registrado en la base de datos, visualizar el mensaje de error.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. 	
Pasos de Ejecución	
<ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado • Clic en el botón Nuevo • Ingresar los Datos requeridos Identificación: 0804344661 Primer nombre: Erika Apellido Paterno: Reina Sexo: Femenino Estado Civil: Soltera Pais: Ecuador • Clic en Grabar Todo/ Grabar 	
Resultado Esperado:	
<ul style="list-style-type: none"> • “Empleado Duplicado”. 	
Evaluación de la Prueba:	

- Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 2

Tabla 102-3 HU-02 Actualización de Empleados

HISTORIA USUARIO	
Número: HU-02	Nombre Historia de Usuario: Actualizar la información de empleados.
Usuario: Administrador	Sprint Asignado: 02
Fecha Inicio: 22/02/17	Fecha Fin: 23/02/17
Descripción: Como administrador necesito actualizar la información de empleados, para poder tenerla renovada en la base de datos para de esa manera utilizarla cuando sea requerida.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Edición correcta de la información de un empleado seleccionado. – No se actualizan los datos de un empleado seleccionado. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-02 se define en la siguiente tabla.

Tabla 103-3 Tabla de Actividades para HU-02

FECHA	ACTIVIDAD	TIEMPO
22/02/17- 23/02/17	Tarea 1: Desarrollo de la edición de empleados. <ul style="list-style-type: none"> • Creación del método actualizar() en la clase NegocioEmpleado. • Creación del método rederizarActualizarEmpleado() en el ControladorEmpleado • Creación del botón para editar en el formularioEmpleado en la vista EmpleadoAdministración 	16 horas

Realizado por: Erika R. 2017

Tabla 104-3 Tarea de Ingeniería 1 para la HU-02

TAREAS DE INGENIERÍA	
SPRINT: 02	Número de Tarea: 01
Nombre de Historia de Usuario: Actualizar la información de empleados.	
Nombre de la Tarea: Implementación de la actualización de Empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 22/02/17	Fecha Fin: 23/02/17
Descripción <ul style="list-style-type: none"> • Creación del método actualizar() en la clase NegocioEmpleado. • Creación del método rederizarActualizarEmpleado() en el ControladorEmpleado • Creación del botón para editar en el formularioEmpleado en la vista EmpleadoAdministración. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Editar correctamente la información de un empleado seleccionado. • No se puede editar los datos de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 105-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la actualización de Empleados.
Nombre de la Prueba: Editar correctamente la información de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 23/02/17
Descripción: Se editará los datos de un empleado que se seleccione, re direccionará a la lista de empleados.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado • Clic en el icono de editar • Cambiar el Apellido Paterno: Ramón por Toscano • Clic en Guardar Todo/Guardar 	

<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se visualiza el empleado modificado.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 106-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la actualización de Empleados.
Nombre de la Prueba: No se puede editar los datos de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 23/02/17
Descripción: No se podrá Editar la información del empleado que se haya seleccionado.	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • No Debe haber conexión con la base de Datos. • Debe haber empleados registrados en la base de datos. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado • Clic en el icono de editar • Cambiar el Apellido Paterno: Toscano por Savedra • Clic en Guardar Todo/Guardar 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • No Se visualiza el empleado modificado. 	
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 3

Tabla 107-3 HU-03 Lista de empleados

HISTORIA USUARIO	
Número: HU-03	Nombre Historia de Usuario: Listar la información de empleados.
Usuario: Administrador	Sprint Asignado: 03
Fecha Inicio: 24/02/17	Fecha Fin: 27/02/17
Descripción: Como administrador necesito listar la información de empleados, para poder visualizarla cuando sea requerida.	
Pruebas de aceptación: <ul style="list-style-type: none">- Visualización de todos los empleados registrados.- No se puede visualizar los datos de los empleados registrados.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-03 se define en la siguiente tabla.

Tabla 108-3 Tabla de Actividades para HU-03

FECHA	ACTIVIDAD	TIEMPO
24/02/17- 27/02/17	Tarea 1: Desarrollo de la lista de empleados <ul style="list-style-type: none">• Creación del método obtenerEmpleados() en la clase NegocioEmpleado• Almacenar los empleados en la variable listaEmpleado del ControladorEmpleado• Creación del formularioEmpleado y la tablaEmpleado en la vista Empleado	16 horas

Realizado por: Erika R. 2017

Tabla 109-3 Tarea de Ingeniería 1 para la HU-03

TAREAS DE INGENIERÍA	
SPRINT: 02	Número de Tarea: 01
Nombre de Historia de Usuario: Listar la información de empleados.	
Nombre de la Tarea: Implementación de la lista de Empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 24/02/17	Fecha Fin: 27/02/17
Descripción	

- Creación del método obtenerEmpleados() en la clase NegocioEmpleado
- Almacenar los empleados en la variable listaEmpleado del ControladorEmpleado
- Creación del formularioEmpleado y la tablaEmpleado en la vista Empleado

Pruebas de Aceptación:

- Visualizar todos los empleados registrados.
- No aparecen los datos de los empleados registrados.

Realizado por: Erika R. 2017

Tabla 110-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 2.1	Nombre de la Tarea de Ingeniería: Implementación de la lista de Empleados.
Nombre de la Prueba: Visualizar todos los empleados registrados.	
Responsable: Erika Reina	Fecha: 27/02/17
Descripción: Se Visualizará la lista de los empleados registrados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualizan los empleados registrados. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 111-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la lista de Empleados.
Nombre de la Prueba: No aparecen los datos de los empleados registrados.	
Responsable: Erika Reina	Fecha: 27/02/17
Descripción: No se podrá Visualizar la lista de los empleados que se encuentran registrados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • No Deben haber empleados registrados en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado 	
Resultado Esperado: <ul style="list-style-type: none"> • No Se visualizan empleados registrados. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 4

Tabla 112-3 Eliminación de Empleados

HISTORIA USUARIO	
Número: HU-04	Nombre Historia de Usuario: Eliminar la información de empleados.
Usuario: Administrador	Sprint Asignado: 02
Fecha Inicio: 28/02/17	Fecha Fin: 01/03/17
Descripción: Como administrador necesito eliminar la información de empleados, para poder darle de baja a un empleado cuando ya no brinde sus servicios en la fundación.	
Pruebas de aceptación:	

<ul style="list-style-type: none"> - Eliminación correcta de la información de un empleado seleccionado. - No se puede eliminar los datos de un empleado seleccionado.
--

Realizado por: Erika R. 2017

Tabla 113-3 Tarea de Ingeniería 1 para la HU-04

TAREAS DE INGENIERÍA	
SPRINT: 02	Número de Tarea: 01
Nombre de Historia de Usuario: Eliminar la información de empleados.	
Nombre de la Tarea: Implementación de la eliminación de Empleados	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 28/02/17	Fecha Fin: 01/03/17
Descripción <ul style="list-style-type: none"> • Creación del método eliminar() en la clase NegocioEmpleado. • Creación del método eliminar() en el ControladorEmpleado • Creación del wConfirmDialogoEliminarEmpleado y el botón respectivo en la vista Empleado. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Eliminar correctamente la información de un empleado seleccionado. • No se puede eliminar los datos de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 114-3 Prueba de aceptación 1 para tarea de ingeniería 4

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la eliminación de Empleados
Nombre de la Prueba: Eliminar correctamente la información de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 01/03/17
Descripción: Se eliminará (Cambiar de estado habilitado a inhabilitado) los datos de un empleado que se seleccione, re direccionará a la lista de empleados.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	

<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado • Clic en el icono para eliminar • Confirmar la eliminación- Clic en Aceptar
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • No se visualizará el empleado modificado en la lista de empleados. • En la base de datos se visualizará el estado cambiado de habilitado a inhabilitado del empleado seleccionado.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 115-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la eliminación de Empleados
Nombre de la Prueba: No se puede eliminar los datos de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 01/03/17
Descripción: No se podrá eliminar (Cambiar de estado habilitado a inhabilitado) los datos de un empleado que se seleccione.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Empleado • Clic en el icono para eliminar • Confirmar la eliminación- Clic en Aceptar 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se emitirá el mensaje “No se puede eliminar el empleado”. • En la base de datos se visualizará el estado habilitado del empleado seleccionado. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Sprint 3

Historia de Usuario 5

Tabla 116-3 HU-05 Reingreso de empleados

HISTORIA USUARIO	
Número: HU-05	Nombre Historia de Usuario: Reingreso de un empleado
Usuario: Administrador	Sprint Asignado: 03
Fecha Inicio: 02/03/17	Fecha Fin: 03/03/17
Descripción: Como administrador necesito reingresar la información de un empleado que ha sido dado de baja, para poder recontratarlo.	
Pruebas de aceptación: <ul style="list-style-type: none">– Reingreso de empleados correctamente y redireccionar a la página correspondiente.– Reingreso de un empleado ya registrado y emitir el mensaje correspondiente.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-01 se define en la siguiente tabla.

Tabla 117-3 Tabla de Actividades para HU-05

FECHA	ACTIVIDAD	TIEMPO
02/03/17- 03/03/17	Tarea 1: Desarrollo del reingreso de empleados. <ul style="list-style-type: none">• Creación del método crear(), en la clase NegocioEmpleado.• Creación del método reingresoEmpleado() en el ControladorEmpleado• Creación del formularioReingresoEmpleado	16 horas

Realizado por: Erika R. 2017

Tabla 118-3 Tarea de Ingeniería 1 para la HU-05

TAREAS DE INGENIERÍA	
SPRINT: 03	Número de Tarea: 01
Nombre de Historia de Usuario: Reingreso de un empleado	
Nombre de la Tarea: Implementación del reingreso de Empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 02/03/17	Fecha Fin: 03/03/17
Descripción <ul style="list-style-type: none"> • Creación del método crear(), en la clase NegocioEmpleado. • Creación del método reingresoEmpleado() en el ControladorEmpleado • Creación del formularioReingresoEmpleado 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Reingresar a un empleado seleccionado • Introducir un empleado ya registrado y emitir el mensaje correspondiente 	

Realizado por: Erika R. 2017

Tabla 119-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación del reingreso de Empleados.
Nombre de la Prueba: Reingresar a un empleado seleccionado	
Responsable: Erika Reina	Fecha: 03/03/17
Descripción: Realizar el reingreso de un empleado dado de baja en la Fundación.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Reingreso • Clic en el ícono del Empleado que desea reingresar • Acepta la confirmación del reingreso • Visualiza a la lista de empleados activos 	
Resultado Esperado: <ul style="list-style-type: none"> • Empleado reingresado. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Tabla 120-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación del reingreso de Empleados.
Nombre de la Prueba: Reingreso de un empleado ya registrado y emitir el mensaje correspondiente.	
Responsable: Erika Reina	Fecha: 03/03/17
Descripción: Realizar el ingreso de un empleado que ya se encuentre activo en la base de datos, visualizar el mensaje de error.	
Condiciones de Ejecución: <ul style="list-style-type: none">• Debe haber conexión con la base de Datos.	
Pasos de Ejecución <ul style="list-style-type: none">• Ingresar al sistema con las credenciales del Administrador.• Clic en Empleados/Reingreso• Clic en el ícono del Empleado que desea reingresar• Acepta la confirmación del reingreso	
Resultado Esperado: <ul style="list-style-type: none">• Emite mensaje “El empleado se encuentra activo”• Redirecciona a la lista de empleados dados de baja.	
Evaluación de la Prueba: <ul style="list-style-type: none">• Satisfactoria	

Realizado por: Erika R. 2017

Historia de Usuario 6

Tabla 121-3 HU-06 Actualización de Empleados

HISTORIA USUARIO	
Número: HU-06	Nombre Historia de Usuario: Actualizar el departamento de empleados de forma masiva.
Usuario: Administrador	Sprint Asignado: 03
Fecha Inicio: 06/03/17	Fecha Fin: 08/03/17
Descripción: Como administrador necesito actualizar los empleados de un departamento a otro, para poder tener los departamentos actualizados cuando sea requerido.	
Pruebas de aceptación: <ul style="list-style-type: none">- Actualización correcta de un empleado seleccionado a otro departamento.- No se actualiza el empleado seleccionado a otro departamento.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-06 se define en la siguiente tabla.

Tabla 122-3 Tabla de Actividades para HU-06

FECHA	ACTIVIDAD	TIEMPO
06/03/17- 08/03/17	Tarea 1: Desarrollo de la actualización de empleados de un departamento a otro. <ul style="list-style-type: none">• Creación del método actualizar() en la clase NegocioDepartamento.• Creación del método actualizarEmpleadoDepartamento() en el ControladorDepartamento• Creación del Formulario actualizarEmpleadoDepartamento	24 horas

Realizado por: Erika R. 2017

Tabla 123-3 Tarea de Ingeniería 1 para la HU-06

TAREAS DE INGENIERÍA	
SPRINT: 03	Número de Tarea: 01
Nombre de Historia de Usuario: Actualizar el departamento de empleados de forma masiva.	
Nombre de la Tarea: Implementación de la actualización de empleados de un departamento a otro.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 06/03/17	Fecha Fin: 08/03/17
Descripción <ul style="list-style-type: none"> • Creación del método actualizar() en la clase NegocioDepartamento. • Creación del método actualizarEmpleadoDepartamento() en el ControladorDepartamento • Creación del Formulario actualizarEmpleadoDepartamento 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Traslado correctamente de un empleado seleccionado a otro departamento. • No se puede trasladar los datos de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 124-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la actualización de empleados de un departamento a otro.
Nombre de la Prueba: Traslado correctamente de un empleado seleccionado a otro departamento.	
Responsable: Erika Reina	Fecha: 08/03/17
Descripción: Se editará a un empleado que se seleccione transfiriéndolo de un departamento a otro, para poder actualizar masivamente un departamento.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Departamentos/Departamento • Clic en el combo escogemos el departamento de origen: Escuela de Fútbol 	

<ul style="list-style-type: none"> • Clic en el otro combo escogemos el departamento de destino: Escuela de Carpintería • Seleccionamos el empleado en el cuadro posterior izquierdo que deseamos actualizar. • Clic en actualizar
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza la lista de los empleados y el departamento al que pertenecen.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 125-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Historia de Usuario: Actualizar el departamento de empleados de forma masiva.
Nombre de la Prueba: No se puede editar los datos de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 08/03/17
Descripción: No se puede trasladar los datos de un empleado seleccionado.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • No Debe haber conexión con la base de Datos. • Debe haber empleados registrados en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Departamentos/Departamento • Clic en el combo escogemos el departamento de origen: Escuela de Fútbol • Clic en el otro combo escogemos el departamento de destino: Escuela de Carpintería • Seleccionamos el empleado en el cuadro posterior izquierdo que deseamos actualizar. • Clic en actualizar 	
Resultado Esperado: <ul style="list-style-type: none"> • No se visualiza la lista de los empleados y el departamento al que pertenecen. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Historia de Usuario 7

Tabla 126-3 HU-07 Lista de cumpleaños del mes

HISTORIA USUARIO	
Número: HU-07	Nombre Historia de Usuario: Generar lista de cumpleaños del mes
Usuario: Administrador	Sprint Asignado: 03
Fecha Inicio: 09/03/17	Fecha Fin: 13/03/17
Descripción: Como administrador necesito listar los cumpleaños del mes, para poder visualizarla cuando sea requerida.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Visualización de todos los cumpleaños del mes. – No se puede visualizar los cumpleaños del mes. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-07 se define en la siguiente tabla.

Tabla 127-3 Tabla de Actividades para HU-07

FECHA	ACTIVIDAD	TIEMPO
09/03/17- 13/03/17	Tarea 1: Desarrollo de la lista de cumpleaños del mes. <ul style="list-style-type: none"> • Creación del método obtenerCumpleaños() en la clase NegocioEmpleado • Almacenar los empleados en la variable listaCumpleaños del ControladorEmpleado • Creación del formularioCumpleañosMes. 	24 horas

Realizado por: Erika R. 2017

Tabla 128-3 Tarea de Ingeniería 1 para la HU-07

TAREAS DE INGENIERÍA	
SPRINT: 03	Número de Tarea: 01
Nombre de Historia de Usuario: Generar lista de cumpleaños del mes.	
Nombre de la Tarea: Implementación de la lista de cumpleaños del mes.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 09/03/17	Fecha Fin: 13/03/17
Descripción <ul style="list-style-type: none"> • Creación del método obtenerCumpleaños() en la clase NegocioEmpleado • Almacenar los empleados en la variable listaCumpleaños del ControladorEmpleado • Creación del formularioCumpleañosMes. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Visualizar la lista de cumpleaños del mes. • No se visualiza la lista de cumpleaños del mes. 	

Realizado por: Erika R. 2017

Tabla 129-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la lista de cumpleaños del mes.
Nombre de la Prueba: Visualizar la lista de cumpleaños del mes.	
Responsable: Erika Reina	Fecha: 13/03/17
Descripción: Se visualizará la lista de los cumpleaños del mes para poder realizar acciones posteriores a esto.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Cumpleaños del mes 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualizan los cumpleaños del mes. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 130-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la lista de cumpleaños del mes.
Nombre de la Prueba: No se visualiza la lista de cumpleaños del mes.	
Responsable: Erika Reina	Fecha: 13/03/17
Descripción: No se podrá visualizar la lista de los cumpleaños del mes.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • No Deben haber empleados registrados en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Cumpleaños del mes 	
Resultado Esperado: <ul style="list-style-type: none"> • No se visualizan cumpleaños del mes, aunque si existan. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 8

Tabla 131-3 HU-08 notificación de felicitación de cumpleaños

HISTORIA USUARIO	
Número: HU-08	Nombre Historia de Usuario: Enviar notificación de felicitación de cumpleaños a cada email.
Usuario: Administrador	Sprint Asignado: 03
Fecha Inicio: 14/03/17	Fecha Fin: 15/03/17
Descripción: Como administrador necesito enviar una notificación a los cumpleaños del mes, para que exista una constancia del acontecimiento.	
Pruebas de aceptación: <ul style="list-style-type: none"> – Que se enví correctamente al correo la notificación de cumpleaños. – Que no se enví correctamente al correo la notificación de cumpleaños. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-08 se define en la siguiente tabla.

Tabla 132-3 Tabla de Actividades para HU-08

FECHA	ACTIVIDAD	TIEMPO
14/03/17- 15/03/17	<p>Tarea 1: Desarrollo de envío de notificación de cumpleaños.</p> <ul style="list-style-type: none"> • Creación del método notificacionCumpleaños() en la clase NegocioEmpleado • Creación del método notificacionCumpleaños() en el ControladorEmpleado • Creación del botón EnviarNotificacion. 	16 horas

Realizado por: Erika R. 2017

Tabla 133-3 Tarea de Ingeniería 1 para la HU-08

TAREAS DE INGENIERÍA	
SPRINT: 03	Número de Tarea: 01
Nombre de Historia de Usuario: Enviar notificación de felicitación de cumpleaños a cada email.	
Nombre de la Tarea: Desarrollo de envío de notificación de cumpleaños.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 14/03/17	Fecha Fin: 15/03/17
<p>Descripción</p> <ul style="list-style-type: none"> • Creación del método notificacionCumpleaños() en la clase NegocioEmpleado • Creación del método notificacionCumpleaños() en el ControladorEmpleado • Creación del botón EnviarNotificacion. 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Enviar correctamente al correo la notificación de cumpleaños. • No se envía al correo la notificación de cumpleaños. 	

Realizado por: Erika R. 2017

Tabla 134-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Desarrollo de envío de notificación de cumpleaños.
Nombre de la Prueba: Enviar correctamente al correo la notificación de cumpleaños.	
Responsable: Erika Reina	Fecha: 15/03/17
Descripción: Se enviará la notificación de cumpleaños al respectivo correo del empleado para tener una constancia de esto.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben cumplir años en el mes presente. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Cumpleaños del mes • Clic en el botón Enviar Notificación 	
Resultado Esperado: <ul style="list-style-type: none"> • Se envió correctamente la notificación de cumpleaños. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 135-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Desarrollo de envío de notificación de cumpleaños.
Nombre de la Prueba: No se envía al correo la notificación de cumpleaños.	
Responsable: Erika Reina	Fecha: 15/03/17
Descripción: No se enviará la notificación de cumpleaños al correo electrónico	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • No debe haber cumpleaños en el mes presente. 	

<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Cumpleaños del mes • Clic en el botón Enviar Notificación
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • No se visualizan cumpleaños del ya que no existen.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Sprint 4

Historia de Usuario 9

Tabla 136-3 HU-09 Ingreso de datos biométricos

HISTORIA USUARIO	
Número: HU-09	Nombre Historia de Usuario: Ingreso de datos biométricos del empleado.
Usuario: Administrador	Sprint Asignado: 04
Fecha Inicio: 16/03/17	Fecha Fin: 22/03/17
Descripción: Como administrador necesito ingresar la información biométrica de los empleados, para poder tenerla almacenada y actualizada en la base de datos y de esa manera utilizarla cuando sea requerida.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Ingreso de datos biométricos correctamente – Introducir datos biométricos de un empleado ya registrado y emitir el mensaje correspondiente. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-09 se define en la siguiente tabla.

Tabla 137-3 Tabla de Actividades para HU-09

FECHA	ACTIVIDAD	TIEMPO
16/03/17- 22/03/17	Tarea 1: Desarrollo del ingreso de datos biométricos de un empleado.	40 horas

	<ul style="list-style-type: none"> • Creación de la clase <code>NegocioBiometria</code> y el método <code>crear()</code>, en ella. • Creación del método <code>rederizarNuevaBiometria()</code> en el <code>ControladorEmpleado</code> • Creación del formulario <code>BiometriaEmpleado</code> 	
--	--	--

Realizado por: Erika R. 2017

Tabla 138-3 Tarea de Ingeniería 1 para la HU-09

TAREAS DE INGENIERÍA	
SPRINT: 04	Número de Tarea: 01
Nombre de Historia de Usuario: Ingreso de datos biométricos del empleado	
Nombre de la Tarea: Implementación del ingreso de datos biométricos de un empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 16/03/17	Fecha Fin: 22/03/17
Descripción <ul style="list-style-type: none"> • Creación de la clase <code>NegocioBiometría</code> y el método <code>crear()</code>, en ella. • Creación del método <code>rederizarNuevaBiometria()</code> en el <code>ControladorEmpleado</code> • Creación del formulario <code>BiometriaEmpleado</code> 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Introducir los datos biométricos correctamente. • Introducir datos biométricos de un empleado ya registrado y emitir el mensaje correspondiente. 	

Realizado por: Erika R. 2017

Tabla 139-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación del ingreso de datos biométricos de un empleado.
Nombre de la Prueba: Introducir los datos biométricos correctamente.	
Responsable: Erika Reina	Fecha: 22/03/17
Descripción: Realizar el ingreso de las imágenes biométricas de las huellas dactilares índices y pulgares para tener su registro almacenado.	
Condiciones de Ejecución:	

<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos.
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Biometría • Clic en el ícono junto al registro del empleado Erika Reina • Ingresar los Datos requeridos Cargar Imagen: Pulgar Derecho Cargar Imagen: Pulgar Izquierdo Cargar Imagen: Índice Derecho Cargar Imagen: Índice Izquierdo • Clic en Grabar Todo/ Grabar
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Datos Biométricos ingresados.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 140-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación del ingreso de datos biométricos de un empleado.
Nombre de la Prueba: Introducir datos biométricos de un empleado ya registrado y emitir el mensaje correspondiente.	
Responsable: Erika Reina	Fecha: 22/03/17
Descripción: Realizar el ingreso de las huellas dactilares de un empleado que ya se encuentre registrado en la base de datos, visualizar el mensaje de error.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Biometría • Clic en el ícono junto al registro del empleado Erika Reina • Ingresar los Datos requeridos Cargar Imagen: Pulgar Derecho Cargar Imagen: Pulgar Izquierdo Cargar Imagen: Índice Derecho Cargar Imagen: Índice Izquierdo 	

<ul style="list-style-type: none"> • Clic en Grabar Todo/ Grabar
Resultado Esperado: <ul style="list-style-type: none"> • “Empleado Duplicado”.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Historia de Usuario 10

Tabla 141-3 HU-10 Actualización de datos biométricos

HISTORIA USUARIO	
Número: HU-10	Nombre Historia de Usuario: Actualizar datos biométricos del empleado.
Usuario: Administrador	Sprint Asignado: 04
Fecha Inicio: 23/03/17	Fecha Fin: 24/03/17
Descripción: Como administrador necesito actualizar la información biométrica de los empleados, para poder tenerla renovada en la base de datos y de esa manera utilizarla cuando sea requerida.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Edición correcta de la información biométrica de un empleado seleccionado. – No se actualizan los datos biométricos de un empleado seleccionado. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-10 se define en la siguiente tabla.

Tabla 142-3 Tabla de Actividades para HU-10

FECHA	ACTIVIDAD	TIEMPO
23/03/17- 24/03/17	Tarea 1: Desarrollo de la edición de datos biométricos de los empleados. <ul style="list-style-type: none"> • Creación del método actualizar() en la clase NegocioBiometria. • Creación del método 	16 horas

	<p>rederizarActualizarBiometria() en el ControladorEmpleado</p> <ul style="list-style-type: none"> • Creación del botón para editar en el formularioBiometriaEmpleado. 	
--	---	--

Realizado por: Erika R. 2017

Tabla 143-3 Tarea de Ingeniería 1 para la HU-10

TAREAS DE INGENIERÍA	
SPRINT: 04	Número de Tarea: 01
Nombre de Historia de Usuario: Actualizar datos biométricos del empleado.	
Nombre de la Tarea: Implementación de la edición de datos biométricos de los empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 23/03/17	Fecha Fin: 24/03/17
<p>Descripción</p> <ul style="list-style-type: none"> • Creación del método actualizar() en la clase NegocioBiometria. • Creación del método rederizarActualizarBiometria() en el ControladorEmpleado • Creación del botón para editar en el formularioBiometriaEmpleado. 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Editar correctamente la información biométrica de un empleado seleccionado. • No se puede editar los datos biométricos de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 144-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la edición de datos biométricos de los empleados.
Nombre de la Prueba: Editar correctamente la información biométrica de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 24/03/17
Descripción: Se editará los datos biométricos de un empleado que se seleccione, para tener la información actualizada.	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Biometría 	

<ul style="list-style-type: none"> • Clic en el ícono junto al registro del empleado Erika Reina • Ingresar los Datos requeridos Cargar Imagen: Pulgar Derecho Nuevo 1 Cargar Imagen: Pulgar Izquierdo Cargar Imagen: Índice Derecho Cargar Imagen: Índice Izquierdo • Clic en Grabar Todo/ Grabar
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza los datos del empleado modificado.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 145-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la edición de datos biométricos de los empleados.
Nombre de la Prueba: No se puede editar los datos biométricos de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 24/03/17
Descripción: No se podrá Editar la información de las huellas dactilares del empleado que se haya seleccionado.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • No Debe haber conexión con la base de Datos. • Debe haber empleados registrados en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Biometría • Clic en el ícono junto al registro del empleado Erika Reina • Ingresar los Datos requeridos Cargar Imagen: Pulgar Derecho Nuevo 2 Cargar Imagen: Pulgar Izquierdo Cargar Imagen: Índice Derecho Cargar Imagen: Índice Izquierdo • Clic en Grabar Todo/ Grabar 	

<p>Resultado Esperado:</p> <ul style="list-style-type: none"> No Se visualiza los datos biométricos del empleado modificado.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> Satisfactoria

Realizado por: Erika R. 2017

Historia de Usuario 11

Tabla 146-3 HU-11 Lista biometrías de empleados

HISTORIA USUARIO	
Número: HU-11	Nombre Historia de Usuario: Listar biometrías de los empleados.
Usuario: Administrador	Sprint Asignado: 04
Fecha Inicio: 27/03/17	Fecha Fin: 28/03/17
Descripción: Como administrador necesito listar la información biométrica de empleados, para poder visualizarla y manipularla cuando sea requerida.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> Visualización de todos los empleados registrados con sus biometrías. No se puede visualizar los datos biométricos de los empleados registrados. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-11 se define en la siguiente tabla.

Tabla 147-3 Tabla de Actividades para HU-11

FECHA	ACTIVIDAD	TIEMPO
27/03/17- 28/03/17	<p>Tarea 1: Desarrollo de la lista de empleados con sus datos biométricos.</p> <ul style="list-style-type: none"> Creación del método obtenerBiometrias() en la clase NegocioBiometria Almacenar las biometrias en la variable listaBiometria del ControladorBiometria Creación del formularioBiometria y la tablaBiometria en la vista Biometria 	16 horas

Realizado por: Erika R. 2017

Tabla 148-3 Tarea de Ingeniería 1 para la HU-11

TAREAS DE INGENIERÍA	
SPRINT: 02	Número de Tarea: 01
Nombre de Historia de Usuario: Listar biometrías de los empleados.	
Nombre de la Tarea: Implementación de la lista de empleados con sus datos biométricos.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 27/03/17	Fecha Fin: 28/03/17
Descripción <ul style="list-style-type: none"> • Creación del método obtenerBiometrias() en la clase NegocioBiometria • Almacenar las biometrías en la variable listaBiometria del ControladorBiometria • Creación del formularioBiometria y la tablaBiometria en la vista Biometria 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Visualizar todos los empleados registrados con sus biometrías. • No se puede visualizar los datos biométricos de los empleados registrados. 	

Realizado por: Erika R. 2017

Tabla 149-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la lista de empleados con sus datos biométricos.
Nombre de la Prueba: Visualizar todos los empleados registrados con sus biometrías.	
Responsable: Erika Reina	Fecha: 28/03/17
Descripción: Se Visualizará la lista de los empleados registrados con sus datos biométricos en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos los datos biométricos de un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Biometría 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualizan los empleados con sus datos biométricos registrados. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 150-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la lista de empleados con sus datos biométricos.
Nombre de la Prueba: No se puede visualizar los datos biométricos de los empleados registrados.	
Responsable: Erika Reina	Fecha: 28/03/17
Descripción: No se podrá Visualizar la lista de los empleados y sus datos biométricos que se encuentran registrados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • No Deben haber empleados registrados en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Biometría 	
Resultado Esperado: <ul style="list-style-type: none"> • No Se visualizan empleados registrados. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 12

Tabla 151-3 HU-12 Historial Laboral de empleados

HISTORIA USUARIO	
Número: HU-12	Nombre Historia de Usuario: Generar el historial laboral del empleado.
Usuario: Administrador	Sprint Asignado: 04
Fecha Inicio: 29/03/17	Fecha Fin: 29/03/17
Descripción: Como administrador necesito generar el historial laboral del empleado para poder tener un control de su asistencia, entre otros parámetros requeridos.	
Pruebas de aceptación: <ul style="list-style-type: none"> – Generación correcta del historial laboral. – No se puede generar correctamente el historial laboral. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-12 se define en la siguiente tabla.

Tabla 152-3 Tabla de Actividades para HU-04

FECHA	ACTIVIDAD	TIEMPO
29/03/17- 29/03/17	<p>Tarea 1: Desarrollo de la generación del historial laboral de los empleados.</p> <ul style="list-style-type: none"> • Creación del método generacionHistorialLab() en la clase NegocioEmpleado • Creación del método calculoHistorialLab() en el ControladorBiometria • Creación del formularioHistorialLaboral y la tablaHistorial en la vista Empleado 	16 horas

Realizado por: Erika R. 2017

Tabla 153-3 Tabla de Actividades para HU-04

TAREAS DE INGENIERÍA	
SPRINT: 04	Número de Tarea: 01
Nombre de Historia de Usuario: Generar el historial laboral del empleado.	
Nombre de la Tarea: Implementación de la generación del historial laboral de los empleados	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 29/03/17	Fecha Fin: 29/03/17
<p>Descripción</p> <ul style="list-style-type: none"> • Creación del método generacionHistorialLab() en la clase NegocioEmpleado • Creación del método calculoHistorialLab() en el ControladorBiometria • Creación del formularioHistorialLaboral y la tablaHistorial en la vista Empleado 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Generar correctamente el historial laboral de un empleado seleccionado. • No se puede generar el historial laboral de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 154-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la generación del historial laboral de los empleados.
Nombre de la Prueba: Generar correctamente el historial laboral de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 29/03/17
Descripción: Se Visualizará el historial laboral de un empleado el cual se generará de datos recopilados diariamente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Historial Laboral • Clic en el ícono al lado derecho de algún empleado seleccionado. 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza el historial laboral de un empleado seleccionado. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 155-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la generación del historial laboral de los empleados.
Nombre de la Prueba: No se puede visualizar los datos biométricos de los empleados registrados.	
Responsable: Erika Reina	Fecha: 29/03/17
Descripción: No se podrá Visualizar la lista de los empleados y sus datos biométricos que se encuentran registrados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado registrado en la base de datos. 	

<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Historial Laboral • Clic en el ícono al lado derecho de algún empleado seleccionado.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • No Se visualiza el historial laboral “Mensaje Advertencia”.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Sprint 5

Historia de Usuario 13

Tabla 156-3 HU-13 Antigüedad Laboral

HISTORIA USUARIO	
Número: HU-13	Nombre Historia de Usuario: Generar la antigüedad laboral del empleado.
Usuario: Administrador	Sprint Asignado: 05
Fecha Inicio: 30/03/17	Fecha Fin: 03/04/17
Descripción: Como administrador necesito generar la antigüedad laboral del empleado para poder tener un control de sus años de trabajo.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Generación correcta de la antigüedad laboral. – No se puede generar correctamente la antigüedad laboral. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-13 se define en la siguiente tabla.

Tabla 157-3 Tabla de Actividades para HU-13

FECHA	ACTIVIDAD	TIEMPO
30/03/17- 03/04/17	Tarea 1: Desarrollo de la generación de la antigüedad laboral de los empleados.	24 horas

	<ul style="list-style-type: none"> • Creación del método generacionAntigüedadLab() en la clase NegocioEmpleado • Creación del método calculoAntigüedadLab() en el ControladorBiometria • Creación del formularioAntigüedadLaboral y la tablaAntigüedad en la vista Empleado 	
--	--	--

Realizado por: Erika R. 2017

Tabla 158-3 Tarea de Ingeniería 1 para la HU-13

TAREAS DE INGENIERÍA	
SPRINT: 05	Número de Tarea: 01
Nombre de Historia de Usuario: Generar la antigüedad laboral del empleado.	
Nombre de la Tarea: Implementación de la generación de la antigüedad laboral de los empleados	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 30/03/17	Fecha Fin: 03/04/17
Descripción	
<ul style="list-style-type: none"> • Creación del método generacionAntigüedadLab() en la clase NegocioEmpleado • Creación del método calculoAntigüedadLab() en el ControladorBiometria • Creación del formularioAntigüedadLaboral y la tablaAntigüedad en la vista Empleado 	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Generar correctamente la antigüedad laboral de un empleado seleccionado. • No se puede generar la antigüedad laboral de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 159-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la generación de la antigüedad laboral de los empleados.
Nombre de la Prueba: Generar correctamente la antigüedad laboral de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 03/04/17
Descripción: Se Visualizará la antigüedad laboral de un empleado el cual se generará de datos recopilados durante su tiempo de trabajo.	

<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado registrado en la base de datos.
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Antigüedad Laboral • Clic en el ícono superior generar.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se visualiza la antigüedad laboral de un empleado seleccionado.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 160-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la generación de la antigüedad laboral de los empleados
Nombre de la Prueba: No se puede generar la antigüedad laboral de un empleado seleccionado,	
Responsable: Erika Reina	Fecha: 03/04/17
Descripción: No se podrá generar la antigüedad laboral de un empleado recién ingresado que se encuentre registrado en la base de datos.	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • No hay empleados registrados 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Antigüedad Laboral • Clic en Generar 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • No Se visualiza la antigüedad laboral “Mensaje Advertencia”. 	
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de usuario 14

Tabla 161-3 HU-14 Actualización Historial Laboral de empleados

HISTORIA USUARIO	
Número: HU-14	Nombre Historia de Usuario: Actualizar el historial laboral del empleado.
Usuario: Administrador	Sprint Asignado: 05
Fecha Inicio: 04/03/17	Fecha Fin: 06/03/17
Descripción: Como administrador necesito actualizar el historial laboral del empleado para poder modificar algunos parámetros de su control de asistencia.	
Pruebas de aceptación: <ul style="list-style-type: none">– Actualización correcta del historial laboral.– No se puede actualizar correctamente el historial laboral.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-14 se define en la siguiente tabla.

Tabla 162-3 Tabla de Actividades para HU-14

FECHA	ACTIVIDAD	TIEMPO
04/03/17- 06/03/17	Tarea 1: Desarrollo de la edición del historial laboral de los empleados. <ul style="list-style-type: none">• Creación del método <code>actualizacionHistorialLab()</code> en la clase <code>NegocioEmpleado</code>• Creación del método <code>actualizarHistorialLab()</code> en el <code>ControladorBiometria</code>• Creación del formulario <code>HistorialLaboral</code> y la tabla <code>Historial</code> en la vista <code>Empleado</code>	24 horas

Realizado por: Erika R. 2017

Tabla 163-3 Tabla de Actividades para HU-14

TAREAS DE INGENIERÍA	
SPRINT: 05	Número de Tarea: 01
Nombre de Historia de Usuario: Edición del historial laboral del empleado.	
Nombre de la Tarea: Implementación de la edición del historial laboral de los empleados	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 04/03/17	Fecha Fin: 06/03/17
Descripción <ul style="list-style-type: none"> • Creación del método actualizaciónHistorialLab() en la clase NegocioEmpleado • Creación del método actualizarHistorialLab() en el ControladorBiometria • Creación del formularioHistorialLaboral y la tablaHistorial en la vista Empleado 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Editar correctamente el historial laboral de un empleado seleccionado. • No se puede editar el historial laboral de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 164-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la edición del historial laboral de los empleados
Nombre de la Prueba: Editar correctamente el historial laboral de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 06/03/17
Descripción: Se podrá editar el historial laboral de un empleado el cual se generará de datos recopilados diariamente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado con su historial laboral registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Historial Laboral • Clic en el ícono editar al lado derecho de algún empleado seleccionado. • Cambiar Total días de vacación: 25 a 27 	
Resultado Esperado: <ul style="list-style-type: none"> • Se edita correctamente el historial laboral de un empleado seleccionado. 	

Evaluación de la Prueba: <ul style="list-style-type: none"> Satisfactoria

Realizado por: Erika R. 2017

Tabla 165-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la edición del historial laboral de los empleados
Nombre de la Prueba: No se puede editar el historial laboral de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 06/03/17
Descripción: No se podrá editar el historial laboral de un empleado que se encuentran registrado en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> Debe haber conexión con la base de Datos. Debe haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> Ingresar al sistema con las credenciales del Administrador. Clic en Empleados/Historial Laboral Clic en el ícono editar al lado derecho de algún empleado seleccionado. Cambiar Total días de vacación: 	
Resultado Esperado: <ul style="list-style-type: none"> No se puede dejar el campo vacío. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 15

Tabla 166-3 HU-15 Sincronizar los registros diarios del empleado

HISTORIA USUARIO	
Número: HU-15	Nombre Historia de Usuario: Sincronizar los registros diarios del empleado.
Usuario: Administrador	Sprint Asignado: 05
Fecha Inicio: 07/04/17	Fecha Fin: 10/04/17
Descripción: Como administrador necesito sincronizar todos los registros diarios de los empleados, para poder tener un control total del mismo y de esa manera utilizarlo cuando sea	

requerido.
<p>Pruebas de aceptación:</p> <ul style="list-style-type: none"> – Verificar que se listen los registros diarios sincronizados desde el lector de huellas digitales. – No se visualiza la sincronización de los registros diarios del empleado.

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-15 se define en la siguiente tabla.

Tabla 167-3 Tabla de Actividades para HU-15

FECHA	ACTIVIDAD	TIEMPO
07/04/17- 10/04/17	<p>Tarea 1: Desarrollo de la sincronización de registros diarios de un empleado.</p> <ul style="list-style-type: none"> • Creación de la clase NegocioRegistrosSinc y el método crear(), en ella. • Creación del método generarRegistrosDiariosSinc() en el ControladorRegistros. • Creación del formularioRegistrosDiariosSinc 	16 horas

Realizado por: Erika R. 2017

Tabla 168-3 Tarea de Ingeniería 1 para la HU-15

TAREAS DE INGENIERÍA	
SPRINT: 05	Número de Tarea: 01
Nombre de Historia de Usuario: Sincronizar los registros diarios del empleado.	
Nombre de la Tarea: Implementación de la sincronización de registros diarios de un empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 07/04/17	Fecha Fin: 10/04/17
<p>Descripción</p> <ul style="list-style-type: none"> • Creación de la clase NegocioRegistrosSinc y el método crear(), en ella. • Creación del método generarRegistrosDiariosSinc() en el ControladorRegistros. • Creación del formularioRegistrosDiariosSinc 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Sincronizar los registros diarios correctamente. • No se sincronizó los datos diarios de un empleado 	

Realizado por: Erika R. 2017

Tabla 169-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la sincronización de registros diarios de un empleado.
Nombre de la Prueba: Introducir los registros diarios correctamente.	
Responsable: Erika Reina	Fecha: 10/04/17
Descripción: Se realizará el traslado de los registros diarios del lector de huellas digitales a la base de datos del sistema.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Iniciar el sistema con un perfil de recursos humanos al inicio de día • Verificar que se visualice la pantalla de sincronización • Al culminar la sincronización, verificar que se visualice la pantalla de registros diarios • Verificar que en la lista cuente con los registros diarios de entrada o salida de los empleados 	
Resultado Esperado: <ul style="list-style-type: none"> • Se sincronizó los registros diarios correctamente. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 170-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la sincronización de registros diarios de un empleado.
Nombre de la Prueba: No se sincronizó los datos diarios de un empleado	
Responsable: Erika Reina	Fecha: 10/04/17
Descripción: Realizar la verificación de que no se sincronizan los registros diarios de un empleado seleccionado.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. 	

<ul style="list-style-type: none"> No debe haber conexión con el lector de huellas digitales.
Pasos de Ejecución <ul style="list-style-type: none"> Iniciar el sistema con un perfil de recursos humanos al inicio de día Verificar que se visualice la pantalla de sincronización Al culminar la sincronización, verificar que se visualice la pantalla de registros diarios Verificar que en la lista cuente con los registros diarios de entrada o salida de los empleados
Resultado Esperado: <ul style="list-style-type: none"> “No se visualiza la sincronización de los registros diarios”.
Evaluación de la Prueba: <ul style="list-style-type: none"> Satisfactoria

Realizado por: Erika R. 2017

Historia de Usuario 16

Tabla 171-3 HU-16 Lista de Registro diario de empleados

HISTORIA USUARIO	
Número: HU-16	Nombre Historia de Usuario: Listar registros diarios de los empleados.
Usuario: Administrador	Sprint Asignado: 03
Fecha Inicio: 11/04/17	Fecha Fin: 12/04/17
Descripción: Como administrador necesito listar los registros diarios de empleados, para poder visualizarlos y manipularlos cuando sea requeridos.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> Visualización de todos los empleados registrados con sus registros diarios. No se puede visualizar los empleados registrados. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-16 se define en la siguiente tabla.

Tabla 172-3 Tabla de Actividades para HU-16

FECHA	ACTIVIDAD	TIEMPO
11/04/17-12/04/17	Tarea 1: Desarrollo de la lista de empleados con sus registros diarios. <ul style="list-style-type: none"> Creación del método 	16 horas

	<p>obtenerRegistrosDiarios() en la clase NegocioRegistrosDiarios</p> <ul style="list-style-type: none"> • Almacenar los registros en la variable listaRegistros del ControladorRegistros • Creación del formularioRegistros y la tablaRegistro en la vista Registro 	
--	---	--

Realizado por: Erika R. 2017

Tabla 173-3 Tarea de Ingeniería 1 para la HU-16

TAREAS DE INGENIERÍA	
SPRINT: 05	Número de Tarea: 01
Nombre de Historia de Usuario: Listar registros diarios de los empleados.	
Nombre de la Tarea: Implementación de la lista de empleados con sus registros diarios.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 11/04/17	Fecha Fin: 12/04/17
Descripción <ul style="list-style-type: none"> • Creación del método obtenerRegistrosDiarios() en la clase NegocioRegistrosDiarios • Almacenar los registros en la variable listaRegistros del ControladorRegistros • Creación del formularioRegistros y la tablaRegistro en la vista Registro 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Visualizar todos los empleados con sus registros diarios. • No se puede visualizar los datos diarios de los empleados registrados. 	

Realizado por: Erika R. 2017

Tabla 174-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la lista de empleados con sus registros diarios.
Nombre de la Prueba: Visualizar todos los empleados con sus registros diarios.	
Responsable: Erika Reina	Fecha: 12/04/17
Descripción: Se Visualizará la lista de los empleados registrados con la opción de conocer sus registros diarios.	

<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos los registros diarios de al menos un empleado registrado en la base de datos.
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Registro Diario • Clic en el ícono derecho del empleado seleccionado
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se visualizan los empleados con sus los datos de sus registros diarios.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 175-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la lista de empleados con sus registros diarios.
Nombre de la Prueba: No se puede visualizar los datos diarios de los empleados registrados.	
Responsable: Erika Reina	Fecha: 12/04/17
Descripción: No se podrá Visualizar la lista de los empleados y sus registros diarios.	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • No Deben haber empleados registrados en la base de datos. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Registro Diario 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • No Se visualizan empleados registrados. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Sprint 6

Historia de usuario 17

Tabla 176-3 HU-17 Actualización de registros diarios de Empleados

HISTORIA USUARIO	
Número: HU-17	Nombre Historia de Usuario: Generar los días de vacaciones a partir del registro diario.
Usuario: Administrador	Sprint Asignado: 06
Fecha Inicio: 13/04/17	Fecha Fin: 17/04/17
Descripción: Como administrador necesito generar el reporte diario de registros, todos los días al momento de iniciar el día por parte de recursos humanos.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Verificar que se visualice el numero de días de vacación posterior a la generación diaria. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-17 se define en la siguiente tabla.

Tabla 177-3 Tabla de Actividades para HU-17

FECHA	ACTIVIDAD	TIEMPO
13/04/17- 17/04/17	<p>Tarea 1: Desarrollo de generación de los días de vacaciones a partir del registro diario.</p> <ul style="list-style-type: none"> • Creación del método generarVacaciones() en la clase NegocioGenerarVacaciones. • Creación del método vacacionesRegistroDiario() en el ControladorRegistro • Creación del Formulario vacacionesRegistro 	24 horas

Realizado por: Erika R. 2017

Tabla 178-3 Tarea de Ingeniería 1 para la HU-17

TAREAS DE INGENIERÍA	
SPRINT: 06	Número de Tarea: 01
Nombre de Historia de Usuario: Generar los días de vacaciones a partir del registro diario.	
Nombre de la Tarea: Implementación de la generación de los días de vacaciones a partir del registro diario.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 13/04/17	Fecha Fin: 17/04/17
Descripción <ul style="list-style-type: none"> • Creación del método generarVacaciones() en la clase NegocioGenerarVacaciones. • Creación del método vacacionesRegistroDiario() en el ControladorRegistro. • Creación del Formulario vacacionesRegistro. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Verificación de la visualización del numero de días de vacaciones posterior a la generación diaria. 	

Realizado por: Erika R. 2017

Tabla 179-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de Tarea de Ingeniería: Implementación de la generación de los días de vacaciones a partir del registro diario.
Nombre de la Prueba: Verificación de la visualización del numero de días de vacaciones posterior a la generación diaria.	
Responsable: Erika Reina	Fecha: 17/04/17
Descripción: Se generará un registro diario de los días de vacación de los empleados, en la pantalla solicitada para tener la información correctamente almacenada y presentada.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ol style="list-style-type: none"> 1) Iniciar el sistema con un perfil de recursos humanos al inicio de día 2) Verificar que se visualice la pantalla de generación de días de vacación 3) Al culminar la generación, verificar que se visualice la pantalla historial laboral por periodo. 4) Verificar que la lista cuente en los registros el numero de días de vacaciones correspondientes al periodo vigente y al numero de días laborados. 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza los días de vacaciones de los empleados. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 18

Tabla 180-3 HU-18 Lista de asistencia de empleados

HISTORIA USUARIO	
Número: HU-18	Nombre Historia de Usuario: Generar la asistencia del empleado.
Usuario: Administrador	Sprint Asignado: 06
Fecha Inicio: 18/04/17	Fecha Fin: 20/04/17
Descripción: Como administrador necesito generar la asistencia del empleado a partir del registro diario del empleado para poder saber el nivel de cumplimiento de su jornada de trabajo.	
Pruebas de aceptación: <ul style="list-style-type: none">– Generación correcta de la asistencia.– No se puede generar correctamente la asistencia.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-18 se define en la siguiente tabla.

Tabla 181-3 Tabla de Actividades para HU-18

FECHA	ACTIVIDAD	TIEMPO
18/04/17- 20/04/17	Tarea 1: Desarrollo de la generación de la asistencia de los empleados. <ul style="list-style-type: none">• Creación del método generacionAsistencia() en la clase NegocioControl• Creación del método calculoAsistencia() en el ControladorControl• Creación del formularioAsistencia y la tablaAsistencia en la vista Control	24 horas

Realizado por: Erika R. 2017

Tabla 182-3 Tarea de Ingeniería 1 para la HU-18

TAREAS DE INGENIERÍA	
SPRINT: 06	Número de Tarea: 01
Nombre de Historia de Usuario: Generar la asistencia del empleado.	
Nombre de la Tarea: Implementación de la generación de la asistencia de los empleados	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 18/04/17	Fecha Fin: 20/04/17
Descripción <ul style="list-style-type: none"> • Creación del método generacionAsistencia() en la clase NegocioControl • Creación del método calculoAsistencia() en el ControladorControl • Creación del formularioAsistencia y la tablaAsistencia en la vista Control 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Generar correctamente la asistencia de un empleado seleccionado. • No se puede generar la asistencia de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 183-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la generación de la asistencia de los empleados.
Nombre de la Prueba: Generar correctamente la asistencia de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 20/04/17
Descripción: Se Visualizará la asistencia de un empleado el cual se generará de datos recopilados a partir de su registro diario.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Registro de Asistencia • Clic en el ícono a la derecha de un empleado seleccionado. 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza la asistencia de un empleado seleccionado. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Tabla 184-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la generación de la asistencia de los empleados.
Nombre de la Prueba: No se puede generar la asistencia de un empleado seleccionado,	
Responsable: Erika Reina	Fecha: 20/04/17
Descripción: No se podrá generar la asistencia de un empleado que no se encuentre registrado en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none">• Debe haber conexión con la base de Datos.• No hay empleados registrados	
Pasos de Ejecución <ul style="list-style-type: none">• Ingresar al sistema con las credenciales del Administrador.• Clic en Empleados/Registro de Asistencia.• Clic en el ícono a la derecha de un empleado seleccionado.	
Resultado Esperado: <ul style="list-style-type: none">• No Se visualiza la asistencia “Mensaje Advertencia”.	
Evaluación de la Prueba: <ul style="list-style-type: none">• Satisfactoria	

Realizado por: Erika R. 2017

Historia de usuario 19

Tabla 185-3 HU-19 Lista de atrasos de empleados

HISTORIA USUARIO	
Número: HU-19	Nombre Historia de Usuario: Generar los atrasos del empleado.
Usuario: Administrador	Sprint Asignado: 06
Fecha Inicio: 18/04/17	Fecha Fin: 20/04/17
Descripción: Como administrador necesito generar los atrasos del empleado a partir del registro diario del empleado para poder saber el nivel de incumplimiento de su jornada de trabajo.	
Pruebas de aceptación: <ul style="list-style-type: none">- Generación correcta de los atrasos.- No se puede generar correctamente los atrasos.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-19 se define en la siguiente tabla.

Tabla 186-3 Tabla de Actividades para HU-19

FECHA	ACTIVIDAD	TIEMPO
18/04/17- 20/04/17	Tarea 1: Desarrollo de la generación de los atrasos de los empleados. <ul style="list-style-type: none">• Creación del método generacionAtrasos() en la clase NegocioControl• Creación del método calculoAtrasos() en el ControladorControl• Creación del formularioAtrasos y la tablaAtrasos en la vista Control	24 horas

Realizado por: Erika R. 2017

Tabla 187-3 Tarea de Ingeniería 1 para la HU-19

TAREAS DE INGENIERÍA	
SPRINT: 06	Número de Tarea: 01
Nombre de Historia de Usuario: Generar los atrasos del empleado.	
Nombre de la Tarea: Implementación de la generación de los atrasos de los empleados	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 18/04/17	Fecha Fin: 20/04/17
Descripción <ul style="list-style-type: none"> • Creación del método generacionAtrasos() en la clase NegocioControl • Creación del método calculoAtrasos() en el ControladorControl • Creación del formularioAtrasos y la tablaAtrasos en la vista Control 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Generar correctamente los atrasos de un empleado seleccionado. • No se puede generar los atrasos de un empleado seleccionado. 	

Realizado por: Erika R. 2017

Tabla 188-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación de la generación de los atrasos de los empleados.
Nombre de la Prueba: Generar correctamente los atrasos de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 20/04/17
Descripción: Se Visualizará los atrasos de un empleado el cual se generará de datos recopilados a partir de su registro diario.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado registrado en la base de datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Registro de Atrasos. • Clic en el ícono a la derecha de un empleado seleccionado. 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza los atrasos de un empleado seleccionado. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Tabla 189-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre de la Tarea de Ingeniería: Implementación de la generación de los atrasos de los empleados.
Nombre de la Prueba: No se puede generar los atrasos de un empleado seleccionado,	
Responsable: Erika Reina	Fecha: 20/04/17
Descripción: No se podrá generar los atrasos de un empleado que no se encuentre registrado en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • No hay empleados registrados 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Registro de Atrasos • Clic en el ícono a la derecha de un empleado seleccionado. 	
Resultado Esperado: <ul style="list-style-type: none"> • No Se visualiza los atrasos “Mensaje Advertencia”. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de usuario 20

Tabla 190-3 HU-20 Historial Laboral-Periodo de empleados

HISTORIA USUARIO	
Número: HU-20	Nombre Historia de Usuario: Actualizar el historial laboral del empleado por periodo.
Usuario: Administrador	Sprint Asignado: 06
Fecha Inicio: 26/04/17	Fecha Fin: 26/04/17
Descripción: Como administrador necesito actualizar el historial laboral del empleado para poder modificar algunos parámetros de su control de asistencia de acuerdo a un periodo determinado.	
Pruebas de aceptación: <ul style="list-style-type: none">- Actualización correcta del historial laboral seleccionando un periodo.- No se puede actualizar correctamente el historial laboral.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-20 se define en la siguiente tabla.

Tabla 191-3 Tabla de Actividades para HU-20

FECHA	ACTIVIDAD	TIEMPO
26/04/17- 26/04/17	Tarea 1: Desarrollo de la edición del historial laboral de los empleados por periodo. <ul style="list-style-type: none">• Creación del método <code>actualizacionHistorialLabPeriodo()</code> en la clase <code>NegocioEmpleado</code>• Creación del método <code>actualizarHistorialLabPeriodo()</code> en el <code>ControladorBiometria</code>• Creación del <code>formularioHistorialLaboralPeriodo</code> y la <code>tablaHistorialPeriodo</code> en la vista <code>Empleado</code>	88 horas

Realizado por: Erika R. 2017

Tabla 192-3 Tarea de Ingeniería 1 para la HU-20

TAREAS DE INGENIERÍA	
SPRINT: 06	Número de Tarea: 01
Nombre de Historia de Usuario: Actualizar el historial laboral del empleado por periodo.	
Nombre de la Tarea: Implementación de la edición del historial laboral de los empleados por periodo.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 26/04/17	Fecha Fin: 26/04/17
Descripción <ul style="list-style-type: none"> • Creación del método actualizaciónHistorialLabPeriodo() en la clase NegocioEmpleado • Creación del método actualizarHistorialLabPeriodo() en el ControladorBiometria • Creación del formularioHistorialLaboralPeriodo y la tablaHistorialPeriodo en la vista Empleado 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Editar correctamente el historial laboral de un empleado de acuerdo a un periodo seleccionado. • No se puede generar el historial laboral de un empleado de acuerdo a un periodo seleccionado. 	

Realizado por: Erika R. 2017

Tabla 193-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la edición del historial laboral de los empleados por periodo.
Nombre de la Prueba: Editar correctamente el historial laboral de un empleado de acuerdo a un periodo seleccionado.	
Responsable: Erika Reina	Fecha: 26/04/17
Descripción: Se podrá editar el historial laboral de un empleado el cual se podrá modificar escogiendo un periodo de una lista.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado con su historial laboral registrado en la base de datos. • Debe haber periodos registrados. 	

<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Historial Laboral • Clic en el ícono editar al lado derecho de algún empleado seleccionado. • Escoger un periodo: Abril-Mayo 2017 • Cambiar Total días de vacación: 25 a 27
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se edita correctamente el historial laboral de un empleado seleccionado.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 194-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la generación del historial laboral de los empleados
Nombre de la Prueba: No se puede generar el historial laboral de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 26/04/17
Descripción: No se podrá Visualizar la lista de los empleados que se encuentran registrados en la base de datos.	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber al menos un empleado registrado en la base de datos. • No Debe haber periodos registrados. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/Historial Laboral • Clic en el ícono editar al lado derecho de algún empleado seleccionado. • Escoger un periodo: No existen periodos registrados • Cambiar Total días de vacación: 27 a 25 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • No se puede editar el historial laboral del empleado en el periodo seleccionado 	
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Sprint 7

Historia de Usuario 21

Tabla 195-3 HU-21 Registro de Solicitud de Vacaciones

HISTORIA USUARIO	
Número: HU-21	Nombre Historia de Usuario: Registro de solicitud de vacaciones del empleado.
Usuario: Administrador	Sprint Asignado: 07
Fecha Inicio: 27/04/17	Fecha Fin: 28/04/17
Descripción: Como administrador necesito ingresar la información para solicitar vacaciones de un empleado específico, para poder descontarle de su total de vacaciones.	
Pruebas de aceptación: <ul style="list-style-type: none">– Ingreso de solicitud de vacaciones correctamente.– No tiene días disponibles para solicitud de vacaciones.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-21 se define en la siguiente tabla.

Tabla 196-3 Tabla de Actividades para HU-21

FECHA	ACTIVIDAD	TIEMPO
27/04/17- 28/04/17	Tarea 1: Desarrollo del Registro de solicitud de vacaciones del empleado. <ul style="list-style-type: none">• Creación de la clase NegocioVacacion y el método crear(), en ella.• Creación del método generarVacacion() en el ControladorVacacion.• Creación del formularioVacacion	16 horas

Realizado por: Erika R. 2017

Tabla 197-3 Tarea de Ingeniería 1 para la HU-21

TAREAS DE INGENIERÍA	
SPRINT: 07	Número de Tarea: 01
Nombre de Historia de Usuario: Registro de solicitud de vacaciones del empleado.	
Nombre de la Tarea: Implementación del Registro de solicitud de vacaciones del empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 27/04/17	Fecha Fin: 28/04/17
Descripción <ul style="list-style-type: none"> • Creación de la clase NegocioVacacion y el método crear(), en ella. • Creación del método generarVacacion() en el ControladorVacacion. • Creación del formularioVacacion 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Introducir la solicitud de vacaciones. • No se puede registrar vacaciones. 	

Realizado por: Erika R. 2017

Tabla 198-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación del Registro de solicitud de vacaciones del empleado.
Nombre de la Prueba: Introducir la solicitud de vacaciones.	
Responsable: Erika Reina	Fecha: 28/04/17
Descripción: Se generará la solicitud de vacaciones de un empleado cuando tenga días disponibles, para que pueda descontarlas de las totales correspondientes.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. • Debe tener días de vacaciones disponibles. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en el ícono derecho junto al registro del empleado Erika Reina • Se visualizan los días de vacaciones disponibles. • Clic en generar. 	

<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Solicitud de vacaciones generada correctamente.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 199-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación del ingreso de registros diarios de un empleado.
Nombre de la Prueba: No se puede registrar vacaciones.	
Responsable: Erika Reina	Fecha: 28/04/17
Descripción: Realizar la verificación de días de vacaciones disponibles para generar la solicitud de las mismas.	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. • No debe tener días de vacaciones disponibles. 	
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en el ícono derecho junto al registro del empleado Erika Reina • Se visualizan los días de vacaciones disponibles. • Clic en generar. 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • “No tiene días de vacaciones disponibles”. 	
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 22

Tabla 200-3 HU-22 Actualización de solicitud de vacaciones

HISTORIA USUARIO	
Número: HU-22	Nombre Historia de Usuario: Actualización de solicitud de vacaciones del empleado.
Usuario: Administrador	Sprint Asignado: 07
Fecha Inicio: 01/05/17	Fecha Fin: 02/05/17
Descripción: Como administrador necesito actualizar la información para solicitud de vacaciones de un empleado específico, para poder tener en correcto funcionamiento el total de vacaciones.	
Pruebas de aceptación: <ul style="list-style-type: none">- Edición de solicitud de vacaciones correctamente.- No se puede editar solicitud de vacaciones.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-22 se define en la siguiente tabla.

Tabla 201-3 Tabla de Actividades para HU-22

FECHA	ACTIVIDAD	TIEMPO
01/05/17- 02/05/17	Tarea 1: Desarrollo del Actualización de solicitud de vacaciones del empleado. <ul style="list-style-type: none">• Creación del método actualizar(), en clase NegocioVacacion.• Creación del método actualizarVacacion() en el ControladorVacacion.• Creación del formularioActualizarVacacion	16 horas

Realizado por: Erika R. 2017

Tabla 202-3 Tarea de Ingeniería 1 para la HU-22

TAREAS DE INGENIERÍA	
SPRINT: 07	Número de Tarea: 01
Nombre de Historia de Usuario: Actualización de solicitud de vacaciones del empleado.	
Nombre de la Tarea: Implementación de Actualización de solicitud de vacaciones del empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 01/05/17	Fecha Fin: 02/05/17
Descripción <ul style="list-style-type: none"> • Creación del método actualizar(), en clase NegocioVacacion. • Creación del método actualizarVacacion() en el ControladorVacacion. • Creación del formularioActualizarVacacion 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Editar la solicitud de vacaciones. • No se puede editar vacaciones. 	

Realizado por: Erika R. 2017

Tabla 203-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de Actualización de solicitud de vacaciones del empleado.
Nombre de la Prueba: Editar la solicitud de vacaciones.	
Responsable: Erika Reina	Fecha: 02/05/17
Descripción: Se editará la solicitud de vacaciones de un empleado cuando tenga días disponibles, para que pueda descontarlas de las totales correspondientes.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. • Debe tener días de vacaciones disponibles. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en el ícono derecho junto al Editar del empleado Erika Reina • Se visualizan los días de vacaciones disponibles. 	

<ul style="list-style-type: none"> • Editamos de 25 a 20 • Clic en grabar/grabar todo.
Resultado Esperado: <ul style="list-style-type: none"> • Solicitud de vacaciones editada correctamente.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 204-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de Actualización de solicitud de vacaciones del empleado.
Nombre de la Prueba: No se puede editar vacaciones.	
Responsable: Erika Reina	Fecha: 02/05/17
Descripción: Realizar la edición luego de la verificación de días de vacaciones disponibles para generar la solicitud de las mismas.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. • No debe tener días de vacaciones disponibles. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en el ícono derecho junto al Editar del empleado Erika Reina • Se visualizan los días de vacaciones disponibles. 	
Resultado Esperado: <ul style="list-style-type: none"> • “No tiene días de vacaciones disponibles”. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de usuario 23

Tabla 205-3 HU-23 Anulación de solicitud de vacación de empleados

HISTORIA USUARIO	
Número: HU-23	Nombre Historia de Usuario: Anulación de solicitud de vacaciones del empleado.
Usuario: Administrador	Sprint Asignado: 07
Fecha Inicio: 03/05/17	Fecha Fin: 09/05/17
Descripción: Como administrador necesito anular la solicitud de vacaciones de un empleado específico, para poder realizar una correcta posterior a rectificación.	
Pruebas de aceptación: <ul style="list-style-type: none">- Anulación de solicitud de vacaciones correctamente.- No se puede anular la solicitud de vacaciones.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-23 se define en la siguiente tabla.

Tabla 206-3 Tabla de Actividades para HU-23

FECHA	ACTIVIDAD	TIEMPO
03/05/17- 09/05/17	Tarea 1: Desarrollo del Anulación de solicitud de vacaciones del empleado. <ul style="list-style-type: none">• Creación del método anular(), en clase NegocioVacacion.• Creación del método anularVacacion() en el ControladorVacacion.• Creación del formularioAnularVacacion	40 horas

Realizado por: Erika R. 2017

Tabla 207-3 Tarea de Ingeniería 1 para la HU-23

TAREAS DE INGENIERÍA	
SPRINT: 07	Número de Tarea: 01
Nombre de Historia de Usuario: Anulación de solicitud de vacaciones del empleado.	
Nombre de la Tarea: Implementación de Anulación de solicitud de vacaciones del empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 03/05/17	Fecha Fin: 09/05/17

<p>Descripción</p> <ul style="list-style-type: none"> • Creación del método anular(), en clase NegocioVacacion. • Creación del método anularVacacion() en el ControladorVacacion. • Creación del formularioAnularVacacion
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Anular la solicitud de vacaciones. • No se puede anular la solicitud de vacaciones.

Realizado por: Erika R. 2017

Tabla 208-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la Anulación de solicitud de vacaciones del empleado.
Nombre de la Prueba: Anular la solicitud de vacaciones.	
Responsable: Erika Reina	Fecha: 09/05/17
Descripción: Se anulará la solicitud de vacaciones de un empleado cuando sea requerido volviendo a acumularse a su favor los días anulados para poder descontarlas de las totales correspondientes.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. • Debe tener solicitud de vacaciones activa. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en el ícono Anular derecho junto al empleado Erika Reina • Se visualizan los días de vacaciones disponibles. • Clic en anular. 	
Resultado Esperado: <ul style="list-style-type: none"> • Solicitud de vacaciones anulada correctamente. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 209-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la Anulación de solicitud de vacaciones del empleado.
Nombre de la Prueba: No se puede anular la solicitud de vacaciones.	
Responsable: Erika Reina	Fecha: 09/05/17
Descripción: Realizar la anulación de la solicitud de vacaciones una vez que ya ha sido emitida para volver a tener los días a favor.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en el ícono Anular derecho junto al empleado Erika Reina • Se visualizan los días de vacaciones disponibles. • Clic en anular. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en el ícono derecho junto al Editar del empleado Erika Reina • Se visualizan los días de vacaciones disponibles. 	
Resultado Esperado: <ul style="list-style-type: none"> • “No tiene solicitud de vacaciones disponibles”. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 24

Tabla 210-3 HU-24 Lista de solicitud de vacaciones activas de empleados

HISTORIA USUARIO	
Número: HU-24	Nombre Historia de Usuario: Lista de solicitud de vacaciones activas del empleado.
Usuario: Administrador	Sprint Asignado: 07
Fecha Inicio: 10/05/17	Fecha Fin: 10/05/17
Descripción: Como administrador necesito listar las solicitudes de vacaciones activas de los empleados, para poder tener un mejor control de las mismas.	
Pruebas de aceptación: <ul style="list-style-type: none">- Lista de solicitud de vacaciones activas correctamente.- No se puede listar la solicitud de vacaciones activas.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-24 se define en la siguiente tabla.

Tabla 211-3 Tabla de Actividades para HU-24

FECHA	ACTIVIDAD	TIEMPO
10/05/17- 10/05/17	Tarea 1: Desarrollo del Lista de solicitud de vacaciones activas del empleado. <ul style="list-style-type: none">• Creación del método listar(), en clase NegocioVacacion.• Creación del método listarVacacion() en el ControladorVacacion.• Creación del formularioListarVacacion	8 horas

Realizado por: Erika R. 2017

Tabla 212-3 Tarea de Ingeniería 1 para la HU-24

TAREAS DE INGENIERÍA	
SPRINT: 07	Número de Tarea: 01
Nombre de Historia de Usuario: Lista de solicitud de vacaciones activas del empleado.	
Nombre de la Tarea: Implementación de Lista de solicitud de vacaciones activas del empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.

Fecha de Inicio: 10/05/17	Fecha Fin: 10/05/17
Descripción <ul style="list-style-type: none"> • Creación del método listar(), en clase NegocioVacacion. • Creación del método listarVacacion() en el ControladorVacacion. • Creación del formularioListarVacacion 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Listar la solicitud de vacaciones. • No se puede listar la solicitud de vacaciones. 	

Realizado por: Erika R. 2017

Tabla 213-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la Lista de solicitud de vacaciones activas del empleado.
Nombre de la Prueba: Listar la solicitud de vacaciones activas.	
Responsable: Erika Reina	Fecha: 10/05/17
Descripción: Se listará las solicitudes de vacaciones activas de los empleados cuando sea requerido para poder visualizarlas y realizar acciones de gestión.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. • Debe haber al menos un empleado que haya emitido solicitud de vacaciones. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en Activas 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza la lista de vacaciones activas correctamente. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 214-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la Lista de solicitud de vacaciones activas del empleado.
Nombre de la Prueba: No se puede listar la solicitud de vacaciones activas.	
Responsable: Erika Reina	Fecha: 10/05/17
Descripción: Realizar la Lista de la solicitud de vacaciones para poder llevar un control de las mismas.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe estar registrado el empleado. • No Debe haber empleados que hayan emitido solicitud de vacaciones. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Vacaciones • Clic en Activas 	
Resultado Esperado: <ul style="list-style-type: none"> • “No tiene empleados con solicitud de vacaciones activas”. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Sprint 8

Historia de Usuario 25

Tabla 215-3 HU-25 Generación de días calendario laborables de empleados

HISTORIA USUARIO	
Número: HU-25	Nombre Historia de Usuario: Generación de días calendario laborables del empleado.
Usuario: Administrador	Sprint Asignado: 08
Fecha Inicio: 11/05/17	Fecha Fin: 15/05/17
Descripción: Como administrador necesito generar los días calendario laborables de los empleados, para poder administrar el calendario anual para el cumplimiento de la jornada de trabajo.	
Pruebas de aceptación: <ul style="list-style-type: none">- Generación de días calendario laborables correctamente.- No se puede Generar los días calendario laborables.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-25 se define en la siguiente tabla.

Tabla 216-3 Tabla de Actividades para HU-25

FECHA	ACTIVIDAD	TIEMPO
11/05/17- 15/05/17	Tarea 1: Desarrollo del Generación de días calendario laborable del empleado. <ul style="list-style-type: none">• Creación del método generar(), en clase NegocioCalendario.• Creación del método generarCalendario() en el ControladorCalendario.• Creación del formularioGenerarCalendario	24 horas

Realizado por: Erika R. 2017

Tabla 217-3 Tarea de Ingeniería 1 para la HU-25

TAREAS DE INGENIERÍA	
SPRINT: 08	Número de Tarea: 01
Nombre de Historia de Usuario: Generación de días calendario laborables del empleado.	
Nombre de la Tarea: Implementación de Generación de días calendario laborables del empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 11/05/17	Fecha Fin: 15/05/17
Descripción <ul style="list-style-type: none"> • Creación del método generar(), en clase NegocioCalendario. • Creación del método generarCalendario() en el ControladorCalendario. • Creación del formularioGenerarCalendario 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Generar los días calendario laborables. • No se puede generar los días calendario laborables. 	

Realizado por: Erika R. 2017

Tabla 218-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la generación de días calendario laborables del empleado.
Nombre de la Prueba: Generar los días calendario laborables.	
Responsable: Erika Reina	Fecha: 15/05/17
Descripción: Se Generará los días calendario laborables de los empleados para administrar las jornadas de trabajo durante todo el año días activos y días no activos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Calendario Laborable • Llenamos los siguientes campos: • Escogemos un año: 2017 • Descripción: Días Laborables • Estado: Activo • Clic en Grabar/Grabar todo 	

<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Días calendario laborables generados correctamente.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 219-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la generación de días calendario laborables del empleado.
Nombre de la Prueba: No se puede Generar los días calendario laborables.	
Responsable: Erika Reina	Fecha: 15/05/17
Descripción: Realizar la generación de los días calendario laborables seleccionando de una lista de años.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. 	
Pasos de Ejecución	
<ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Calendario Laborable • Escogemos un año: 2016 	
Resultado Esperado:	
<ul style="list-style-type: none"> • “No se puede escoger un año anterior”. 	
Evaluación de la Prueba:	
<ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 26

Tabla 220-3 HU-26 Actualización de días calendario laborables de empleados

HISTORIA USUARIO	
Número: HU-26	Nombre Historia de Usuario: Actualización de días calendario laborables del empleado.
Usuario: Administrador	Sprint Asignado: 08
Fecha Inicio: 16/05/17	Fecha Fin: 17/05/17
Descripción: Como administrador necesito actualizar los días calendario laborables de los empleados, para poder administrar sin errores el calendario anual que facilitará el cumplimiento de la jornada de trabajo.	
Pruebas de aceptación: <ul style="list-style-type: none">- Actualización de días calendario laborables correctamente.- No se puede actualizar los días calendario laborables.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-26 se define en la siguiente tabla.

Tabla 221-3 Tabla de Actividades para HU-26

FECHA	ACTIVIDAD	TIEMPO
16/05/17- 17/05/17	Tarea 1: Desarrollo de la actualización de días calendario laborables del empleado. <ul style="list-style-type: none">• Creación del método actualizar(), en clase NegocioCalendario.• Creación del método actualizarCalendario() en el ControladorCalendario.• Creación del formularioActualizarCalendario	16 horas

Realizado por: Erika R. 2017

Tabla 222-3 Tarea de Ingeniería 1 para la HU-26

TAREAS DE INGENIERÍA	
SPRINT: 08	Número de Tarea: 01
Nombre de Historia de Usuario: Actualización de días calendario laborables del empleado.	
Nombre de la Tarea: Implementación de Actualización de días calendario laborables del empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 16/05/17	Fecha Fin: 17/05/17
Descripción	
<ul style="list-style-type: none"> • Creación del método actualizar(), en clase NegocioCalendario. • Creación del método actualizarCalendario() en el ControladorCalendario. • Creación del formularioActualizarCalendario 	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Actualizar los días calendario laborables. • No se puede actualizar los días calendario laborables. 	

Realizado por: Erika R. 2017

Tabla 223-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la actualización de días calendario laborables del empleado.
Nombre de la Prueba: Actualizar los días calendario laborables.	
Responsable: Erika Reina	Fecha: 17/05/17
Descripción: Se actualizará los días calendario laborables de los empleados para modificar en caso de algún error las jornadas de trabajo durante todo el año días activos y días no activos.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. 	
Pasos de Ejecución	
<ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Calendario Laborable • Editamos los siguientes campos: • Escogemos un año: 2018 • Descripción: Días Laborables 	

<ul style="list-style-type: none"> • Estado: Inactivo • Clic en Grabar/Grabar todo
Resultado Esperado: <ul style="list-style-type: none"> • Días calendario laborables generados correctamente.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 224-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la Actualización de días calendario laborables del empleado.
Nombre de la Prueba: No se puede Actualizar los días calendario laborables.	
Responsable: Erika Reina	Fecha: 17/05/17
Descripción: Realizar la actualización de un campo del calendario laboral seleccionando para una administración correcta.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Calendario Laborable • Editamos los siguientes campos: • Escogemos un año: 2018 • Descripción: Días Laborables • Estado: Activo • Clic en Grabar/Grabar todo 	
Resultado Esperado: <ul style="list-style-type: none"> • “No se puede escoger activar un año posterior al actual”. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de Usuario 27

Tabla 225-3 HU-27 Lista de días calendario laborables de empleados

HISTORIA USUARIO	
Número: HU-27	Nombre Historia de Usuario: Lista de días calendario laborables del empleado.
Usuario: Administrador	Sprint Asignado: 08
Fecha Inicio: 18/05/17	Fecha Fin: 22/05/17
Descripción: Como administrador necesito listar los registros de calendario laborables de los empleados, para poder visualizarlos cuando sea necesario.	
Pruebas de aceptación: <ul style="list-style-type: none">- Lista de días calendario laborables correctamente.- No se puede listar los días calendario laborables.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-27 se define en la siguiente tabla.

Tabla 226-3 Tabla de Actividades para HU-27

FECHA	ACTIVIDAD	TIEMPO
18/05/17- 22/05/17	Tarea 1: Desarrollo de la lista de días calendario laborables del empleado. <ul style="list-style-type: none">• Creación del método listar(), en clase NegocioCalendario.• Creación del método listarCalendario() en el ControladorCalendario.• Creación del formularioListarCalendario	24 horas

Realizado por: Erika R. 2017

Tabla 227-3 Tarea de Ingeniería 1 para la HU-27

TAREAS DE INGENIERÍA	
SPRINT: 08	Número de Tarea: 01
Nombre de Historia de Usuario: Lista de días calendario laborables del empleado.	
Nombre de la Tarea: Implementación de Lista de días calendario laborables del empleado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 18/05/17	Fecha Fin: 22/05/17
Descripción	

- Creación del método listar(), en clase NegocioCalendario.
- Creación del método listarCalendario() en el ControladorCalendario.
- Creación del formularioListarCalendario

Pruebas de Aceptación:

- Listar los días calendario laborables.
- No se puede listar los días calendario laborables.

Realizado por: Erika R. 2017

Tabla 228-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la lista de días calendario laborables del empleado.
Nombre de la Prueba: Listar los días calendario laborables.	
Responsable: Erika Reina	Fecha: 22/05/17
Descripción: Se listará los días calendario laborables de los empleados para visualizar los datos de acuerdo a un año filtrado.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Calendario Laborable • Filtramos el año 2017 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza correctamente. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 229-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la Lista de días calendario laborables del empleado.
Nombre de la Prueba: No se puede Listar los días calendario laborables.	
Responsable: Erika Reina	Fecha: 22/05/17
Descripción: Realizar la lista de los días calendarios activos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Calendario Laborable • Filtramos el año 2016 	
Resultado Esperado: <ul style="list-style-type: none"> • “No se puede visualizar días calendario activo”. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de usuario 28

Tabla 230-3 HU-28 Lista de días calendario laborables de empleados

HISTORIA USUARIO	
Número: HU-28	Nombre Historia de Usuario: Registro de permisos por tipo.
Usuario: Administrador	Sprint Asignado: 08
Fecha Inicio: 23/05/17	Fecha Fin: 24/05/17
Descripción: Como administrador necesito registrar los permisos de los empleados de acuerdo a un tipo establecido por la fundación para poder realizar la gestión necesaria.	
Pruebas de aceptación: <ul style="list-style-type: none"> – Registro de permisos por un tipo determinado. – No se puede realizar el registro de permisos. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-28 se define en la siguiente tabla.

Tabla 231-3 Tabla de Actividades para HU-28

FECHA	ACTIVIDAD	TIEMPO
23/05/17- 24/05/17	<p>Tarea 1: Desarrollo del registro de permisos por tipo del empleado.</p> <ul style="list-style-type: none"> • Creación del método registroTipo(), en clase NegocioPermisos. • Creación del método registroTipo() en el ControladorCalendario. • Creación del formularioRegistroTipo 	24 horas

Realizado por: Erika R. 2017

Tabla 232-3 Tarea de Ingeniería 1 para la HU-28

TAREAS DE INGENIERÍA	
SPRINT: 08	Número de Tarea: 01
Nombre de Historia de Usuario: Registro de permisos por tipo.	
Nombre de la Tarea: Implementación del registro de permisos por tipo.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 23/05/17	Fecha Fin: 24/05/17
<p>Descripción</p> <ul style="list-style-type: none"> • Creación del método registroTipo(), en clase NegocioPermisos. • Creación del método registroTipo() en el ControladorCalendario. • Creación del formularioRegistroTipo 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Registrar los permisos de un empleado de acuerdo a un tipo requerido • No se puede registrar los permisos de un empleado. 	

Realizado por: Erika R. 2017

Tabla 233-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación del registro de permisos por tipo.
Nombre de la Prueba: Registrar los permisos de un empleado de acuerdo a un tipo requerido.	
Responsable:	Fecha: 24/05/17

Erika Reina	
Descripción: Se guardará para cada empleado los permisos que soliciten para tener un mejor control de los mismos.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. • Debe haber empleados registrados. 	
Pasos de Ejecución	
<ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Permisos • Clic en un empleado • Clic en el ícono registrar junto al empleado seleccionado • Ingresamos los siguientes campos: • Justificación: Por días de Lactancia • Fecha: 03/03/2017 • Estado: Activo • Tipo: Lactancia 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Se registra correctamente. 	
Evaluación de la Prueba:	
<ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 234-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación del registro de permisos por tipo.
Nombre de la Prueba: No se puede registrar los permisos de un empleado.	
Responsable: Erika Reina	Fecha: 24/05/17
Descripción: Realizar el registro de un permiso ya registrado.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. • Debe haber empleados registrados. 	

<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Permisos • Ingresamos los siguientes campos: • Justificación: Por días de Lactancia • Fecha: 03/03/2017 • Estado: Activo • Tipo: Lactancia
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • “Ya se encuentra registrado ese permiso”.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Sprint 9

Historia de usuario 29

Tabla 235-3 HU-29 Lista de días calendario laborables de empleados

HISTORIA USUARIO	
Número: HU-29	Nombre Historia de Usuario: Actualización de permisos de empleados.
Usuario: Administrador	Sprint Asignado: 09
Fecha Inicio: 25/05/17	Fecha Fin: 26/05/17
Descripción: Como administrador necesito actualizar los permisos de los empleados registrados para poder realizar la corrección de algún tipo de error que haya ocurrido.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Actualización de permisos registrados. – No se puede realizar la actualización de permisos. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-29 se define en la siguiente tabla.

Tabla 236-3 Tabla de Actividades para HU-29

FECHA	ACTIVIDAD	TIEMPO
25/05/17- 26/05/17	<p>Tarea 1: Desarrollo de la actualización de permisos registrados de empleados.</p> <ul style="list-style-type: none"> • Creación del método actualizaciónPermiso(), en clase NegocioPermisos. • Creación del método actualizaciónPermiso () en el ControladorCalendario. • Creación del formularioActualizaciónPermiso 	16 horas

Realizado por: Erika R. 2017

Tabla 237-3 Tarea de Ingeniería 1 para la HU-29

TAREAS DE INGENIERÍA	
SPRINT: 09	Número de Tarea: 01
Nombre de Historia de Usuario: Actualización de permisos de empleados.	
Nombre de la Tarea: Implementación de la actualización de permisos registrados de empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 25/05/17	Fecha Fin: 26/05/17
<p>Descripción</p> <ul style="list-style-type: none"> • Creación del método actualizaciónPermiso(), en clase NegocioPermisos. • Creación del método actualizaciónPermiso () en el ControladorCalendario. • Creación del formularioActualizaciónPermiso 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Actualizar los permisos de un empleado registrado. • No se puede actualizar los permisos de un empleado. 	

Realizado por: Erika R. 2017

Tabla 238-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la actualización de permisos registrados de empleados.
Nombre de la Prueba: Actualizar los permisos de un empleado registrado.	
Responsable:	Fecha: 26/05/17

Erika Reina	
Descripción: Se guardará para cada empleado los permisos que soliciten para tener un mejor control de los mismos.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. • Debe haber empleados registrados. 	
Pasos de Ejecución	
<ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Permisos • Clic en un empleado • Clic en el ícono actualizar junto al empleado seleccionado • Ingresamos los siguientes campos: • Justificación: Por Defunción Familiar • Fecha: 03/03/2017 • Estado: Activo • Tipo: Defunción 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Se editó correctamente. 	
Evaluación de la Prueba:	
<ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Tabla 239-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la actualización de permisos registrados de empleados.
Nombre de la Prueba: No se puede actualizar los permisos de un empleado.	
Responsable: Erika Reina	Fecha: 26/05/17
Descripción: No se podrá realizar la actualización de un permiso ya registrado, debido a que debe llenar todos los campos.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. 	

<ul style="list-style-type: none"> • Debe haber empleados registrados.
<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Permisos • Ingresamos los siguientes campos: • Justificación: Por días de Lactancia • Fecha: • Estado: Activo • Tipo: Lactancia
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • “No se puede actualizar el permiso, Debe ingresar todos los campos”.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 30

Tabla 240-3 HU-30 Lista de días calendario laborables de empleados

HISTORIA USUARIO	
Número: HU-30	Nombre Historia de Usuario: Lista de permisos de empleados.
Usuario: Administrador	Sprint Asignado: 09
Fecha Inicio: 29/05/17	Fecha Fin: 30/05/17
Descripción: Como administrador necesito listar los permisos de los empleados registrados para poder visualizar la información cuando sea requerida	
<p>Pruebas de aceptación:</p> <ul style="list-style-type: none"> – Lista de permisos de empleados registrados. – No se puede realizar la lista de permisos. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-30 se define en la siguiente tabla.

Tabla 241-3 Tabla de Actividades para HU-30

FECHA	ACTIVIDAD	TIEMPO
29/05/17- 30/05/17	<p>Tarea 1: Desarrollo de la lista de permisos registrados de empleados.</p> <ul style="list-style-type: none"> • Creación del método listaPermiso(), en clase NegocioPermisos. • Creación del método listaPermiso () en el ControladorCalendario. • Creación del formularioListaPermiso 	16 horas

Realizado por: Erika R. 2017

Tabla 242-3 Tarea de Ingeniería 1 para la HU-30

TAREAS DE INGENIERÍA	
SPRINT: 09	Número de Tarea: 01
Nombre de Historia de Usuario: Lista de permisos de empleados registrados.	
Nombre de la Tarea: Implementación de la lista de permisos registrados de empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 29/05/17	Fecha Fin: 30/05/17
<p>Descripción</p> <ul style="list-style-type: none"> • Creación del método listaPermiso(), en clase NegocioPermisos. • Creación del método listaPermiso () en el ControladorCalendario. • Creación del formularioListaPermiso 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Listar los permisos de un empleado registrado. • No se puede listar los permisos de un empleado. 	

Realizado por: Erika R. 2017

Tabla 243-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la lista de permisos registrados de empleados.
Nombre de la Prueba: Listar los permisos de un empleado registrado.	
Responsable: Erika Reina	Fecha: 30/05/17

Descripción: Se visualizará la lista de permisos registrados filtrando por un tipo.
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. • Debe haber permisos de empleados registrados.
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Permisos. • Introduce un tipo de permiso en la parte superior del dato tipo.
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza los permisos del tipo introducido.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 244-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la lista de permisos registrados de empleados.
Nombre de la Prueba: No se puede listar los permisos de un empleado.	
Responsable: Erika Reina	Fecha: 30/05/17
Descripción: No se podrá visualizar la lista de permisos que no hayan sido registrados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber días calendario ingresados. • Debe haber permisos de empleados registrados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Permisos • Introduce un tipo de permiso no registrado: Por cumpleaños 	
Resultado Esperado: <ul style="list-style-type: none"> • “No se visualizan permisos de ese tipo”. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 31

Tabla 245-3 HU-31 Registro de horarios laborables de empleados

HISTORIA USUARIO	
Número: HU-31	Nombre Historia de Usuario: Registro de horarios laborables.
Usuario: Administrador	Sprint Asignado: 08
Fecha Inicio: 31/05/17	Fecha Fin: 07/06/17
Descripción: Como administrador necesito registrar los horarios laborables de los empleados para poder tener el respaldo de las horas que deben cumplirse diariamente.	
Pruebas de aceptación: <ul style="list-style-type: none">– Registro de horarios laborables.– No se puede realizar el registro de horarios laborables.	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-31 se define en la siguiente tabla.

Tabla 246-3 Tabla de Actividades para HU-31

FECHA	ACTIVIDAD	TIEMPO
31/05/17- 07/06/17	Tarea 1: Desarrollo del registro de horarios laborables de empleados. <ul style="list-style-type: none">• Creación del método registro(), en clase NegocioHorariosLaborables.• Creación del método registro() en el ControladorCalendario.• Creación del formularioRegistroHorario	48 horas

Realizado por: Erika R. 2017

Tabla 247-3 Tarea de Ingeniería 1 para la HU-31

TAREAS DE INGENIERÍA	
SPRINT: 08	Número de Tarea: 01
Nombre de Historia de Usuario: Registro de horarios laborables.	
Nombre de la Tarea: Implementación del registro de horarios laborables de empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 31/05/17	Fecha Fin: 07/06/17
Descripción <ul style="list-style-type: none"> • Creación del método registro(), en clase NegocioHorariosLaborables. • Creación del método registro() en el ControladorCalendario. • Creación del formularioRegistroHorario 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Registrar los horarios laborables de un empleado seleccionado. • No se puede registrar los horarios laborables de un empleado. 	

Realizado por: Erika R. 2017

Tabla 248-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación del registro de horarios laborables de empleados.
Nombre de la Prueba: Registrar los horarios laborables de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 07/06/17
Descripción: Se guardará para cada empleado los horarios laborables que se han establecido para lograr un cumplimiento de los mismos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber empleados registrados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Horarios laborables • Clic en el empleado Erika Reina • Ingresamos los siguientes datos: • Nombre: Matutino • Descripción: En la mañana • Hora de Entrada: 08:00 	

<ul style="list-style-type: none"> • Hora de Almuerzo: 12:00 • Hora de Regreso del Almuerzo: 14:00 • Hora de Salida: 18:00 • Número de horas laborables: 8
Resultado Esperado: <ul style="list-style-type: none"> • Se registra correctamente.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 249-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación del registro de horarios laborables de empleados.
Nombre de la Prueba: No se puede registrar los horarios laborables de un empleado.	
Responsable: Erika Reina	Fecha: 07/06/17
Descripción: Realizar el registro de un horario laboral dejando campos vacíos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber empleados registrados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Horarios laborables • Clic en el empleado Fausto Guaña • Ingresamos los siguientes datos: • Nombre: Matutino • Descripción: En la mañana • Hora de Entrada: 08:00 • Hora de Almuerzo: 12:00 • Hora de Regreso del Almuerzo: 14:00 • Hora de Salida: 18:00 • Número de horas laborables: 	
Resultado Esperado: <ul style="list-style-type: none"> • “Debe llenar todos los campos”. 	

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017

Sprint 10

Historia de usuario 32

Tabla 250-3 HU-32 Actualización de horarios laborables de empleados

HISTORIA USUARIO	
Número: HU-32	Nombre Historia de Usuario: Actualización de horarios laborables.
Usuario: Administrador	Sprint Asignado: 08
Fecha Inicio: 08/06/17	Fecha Fin: 09/06/17
Descripción: Como administrador necesito actualizar los horarios laborables de los empleados para poder tener en correcto mantenimiento las horas que deben cumplirse diariamente por los empleados.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Actualización de horarios laborables. – No se puede realizar la actualización de horarios laborables. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-32 se define en la siguiente tabla.

Tabla 251-3 Tabla de Actividades para HU-32

FECHA	ACTIVIDAD	TIEMPO
08/06/17- 09/06/17	Tarea 1: Desarrollo de la actualización de horarios laborables de empleados. <ul style="list-style-type: none"> • Creación del método actualización(), en clase NegocioHorariosLaborables. • Creación del método actualización() en el ControladorCalendario. • Creación del formularioActualizaciónHorario 	16 horas

Realizado por: Erika R. 2017

Tabla 252-3 Tarea de Ingeniería 1 para la HU-32

TAREAS DE INGENIERÍA	
SPRINT: 08	Número de Tarea: 01
Nombre de Historia de Usuario: Actualización de horarios laborables.	
Nombre de la Tarea: Implementación de la actualización de horarios laborables de empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 08/06/17	Fecha Fin: 09/06/17
Descripción <ul style="list-style-type: none"> • Creación del método actualización(), en clase NegocioHorariosLaborables. • Creación del método actualización() en el ControladorCalendario. • Creación del formularioActualizaciónHorario 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Actualizar los horarios laborables de un empleado seleccionado. • No se puede actualizar los horarios laborables de un empleado. 	

Realizado por: Erika R. 2017

Tabla 253-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la actualización de horarios laborables de empleados.
Nombre de la Prueba: Actualizar los horarios laborables de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 09/06/17
Descripción: Se guardará para cada empleado los horarios laborables que se han establecido para lograr un cumplimiento de los mismos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber empleados registrados. • Debe haber al menos un horario laboral registrado. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Horarios laborables • Clic en el empleado Erika Reina • Clic en el ícono editar • Ingresamos los siguientes datos: 	

<ul style="list-style-type: none"> • Nombre: Matutino • Descripción: En la mañana • Hora de Entrada: 08:00 • Hora de Almuerzo: 12:00 • Hora de Regreso del Almuerzo: 14:00 • Hora de Salida: 17:00 • Número de horas laborables: 7
Resultado Esperado: <ul style="list-style-type: none"> • Se actualizó correctamente.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Tabla 254-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la actualización de horarios laborables de empleados.
Nombre de la Prueba: No se puede actualizar los horarios laborables de un empleado.	
Responsable: Erika Reina	Fecha: 09/06/17
Descripción: Realizar la actualización de un horario laboral dejando campos vacíos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber empleados registrados. • Debe haber al menos un horario laboral registrado. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Horarios laborables • Clic en el empleado Fausto Guaña • Ingresamos los siguientes datos: • Nombre: Matutino • Descripción: En la mañana • Hora de Entrada: 08:00 • Hora de Almuerzo: 12:00 • Hora de Regreso del Almuerzo: 14:00 • Hora de Salida: 	

<ul style="list-style-type: none"> Número de horas laborables: 8
Resultado Esperado: <ul style="list-style-type: none"> “Debe llenar todos los campos”.
Evaluación de la Prueba: <ul style="list-style-type: none"> Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 33

Tabla 255-3 HU-33 Lista de horarios laborables de empleados

HISTORIA USUARIO	
Número: HU-33	Nombre Historia de Usuario: Lista de horarios laborables.
Usuario: Administrador	Sprint Asignado: 08
Fecha Inicio: 12/06/17	Fecha Fin: 13/06/17
Descripción: Como administrador necesito listar los horarios laborables de los empleados para poder visualizarlos y así gestionarlos cuando sea requerido.	
Pruebas de aceptación: <ul style="list-style-type: none"> – Lista de horarios laborables. – No se puede realizar la lista de horarios laborables. 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-33 se define en la siguiente tabla.

Tabla 256-3 Tabla de Actividades para HU-33

FECHA	ACTIVIDAD	TIEMPO
12/06/17- 13/06/17	Tarea 1: Desarrollo de la lista de horarios laborables de empleados. <ul style="list-style-type: none"> • Creación del método lista(), en clase NegocioHorariosLaborables. • Creación del método lista() en el ControladorCalendario. • Creación del formularioListaHorario 	16 horas

Realizado por: Erika R. 2017

Tabla 257-3 Tarea de Ingeniería 1 para la HU-33

TAREAS DE INGENIERÍA	
SPRINT: 08	Número de Tarea: 01
Nombre de Historia de Usuario: Lista de horarios laborables.	
Nombre de la Tarea: Implementación de la lista de horarios laborables de empleados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 12/06/17	Fecha Fin: 13/06/17
Descripción <ul style="list-style-type: none"> • Creación del método lista(), en clase NegocioHorariosLaborables. • Creación del método lista() en el ControladorCalendario. • Creación del formularioListaHorario 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Listar los horarios laborables de un empleado seleccionado. • No se puede listar los horarios laborables de un empleado. 	

Realizado por: Erika R. 2017

Tabla 258-3 Prueba de aceptación 1 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre Tarea de Ingeniería: Implementación de la lista de horarios laborables de empleados.
Nombre de la Prueba: Listar los horarios laborables de un empleado seleccionado.	
Responsable: Erika Reina	Fecha: 13/06/17
Descripción: Se desplegará una lista con los empleados y sus respectivos horarios laborables para de esa manera realizar las diferentes opciones presentadas.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber empleados registrados. • Debe haber al menos un horario laboral registrado. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Horarios laborables 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza la lista de empleados con sus horarios laborables correctamente. 	
Evaluación de la Prueba:	

- Satisfactoria

Realizado por: Erika R. 2017

Tabla 259-3 Prueba de aceptación 2 para tarea de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.2	Nombre Tarea de Ingeniería: Implementación de la lista de horarios laborables de empleados.
Nombre de la Prueba: No se puede listar los horarios laborables de un empleado.	
Responsable: Erika Reina	Fecha: 13/06/17
Descripción: Realizar la lista los horarios laborables pertenecientes a los empleados registrados.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Debe haber empleados registrados. • No debe haber horarios laborables registrados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador. • Clic en Empleados/ Horarios laborables 	
Resultado Esperado: <ul style="list-style-type: none"> • “No se visualizan empleados con horarios laborables registrados”. 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de usuario 34

Tabla 260-3 HU-34 Reporte de empleados clasificados por departamento

HISTORIA USUARIO	
Número: HU-34	Nombre Historia de Usuario: Reporte de empleados clasificados por departamento.
Usuario: Administrador	Sprint Asignado: 10
Fecha Inicio: 14/06/17	Fecha Fin: 15/06/17

Descripción: Como administrador necesito visualizar el reporte de empleados clasificados por departamento para tener un control de la cantidad de ellos que constan en los diferentes departamentos de la fundación.
Pruebas de aceptación: <ul style="list-style-type: none"> – Verificar que se visualice el reporte de empleados en formato PDF

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-34 se define en la siguiente tabla.

Tabla 261-3 Actividades de HU-34

FECHA	ACTIVIDAD	TIEMPO
14/06/17- 15/06/17	<p>Tarea 1: Desarrollo del reporte de empleados clasificados por departamento.</p> <ul style="list-style-type: none"> • Crear el método de consulta obtenerEmpleadoDepartamento en la clase NegocioEmpleado • Creación de la clase ModeloEmpleado • Crear el método obtenerEmpleadoDepartamento en la clase ControladorEmpleado • Crear el método initEmpleadosDepartamento en la clase ControladorEmpleado • Crear el método PDFEmpleadosDepartamento la clase ControladorEmpleado • Diseño de reporte en IReport 	16 horas

Realizado por: Erika R. 2017

Tabla 262-3 Tarea de Ingeniería para la HU-34

TAREAS DE INGENIERÍA	
SPRINT: 09	Número de Tarea: 01
Nombre de Historia de Usuario: Reporte de empleados clasificados por departamento.	
Nombre de la Tarea: Implementación del reporte de empleados clasificados por departamento.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 14/06/17	Fecha Fin: 15/06/17
Descripción	

- Crear el método de consulta obtenerEmpleadoDepartamento en la clase NegocioEmpleado
- Creación de la clase ModeloEmpleado
- Crear el método obtenerEmpleadoDepartamento en la clase ControladorEmpleado
- Crear el método initEmpleadosDepartamento en la clase ControladorEmpleado
- Crear el método PDFEmpleadosDepartamento la clase ControladorEmpleado
- Diseño de reporte en IReport

Pruebas de Aceptación:

- Verificar que se descargue y visualice el reporte de empleados clasificados por departamento.

Realizado por: Erika R. 2017

Tabla 263-3 Prueba de aceptación 1 para tarea de de ingeniería 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación del reporte de empleados clasificados por departamento.
Nombre de la Prueba: Verificar que se descargue y visualice el reporte de empleados clasificados por departamento.	
Responsable: Erika Reina	Fecha: 15/06/17
Descripción: Se descargará el reporte de empleados clasificados por departamento en un archivo formato PDF el mismo que se abrirá en el navegador para su visualización respectiva.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber empleados en departamentos. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador • Clic en Reportes/Empleados clasificados por departamento. • Clic en el reporte PDF descargado • Visualizar la lista de empleados clasificados por departamento. 	
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza el reporte PDF en el navegador respectivo 	
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria 	

Realizado por: Erika R. 2017

Historia de usuario 35

Tabla 264-3 HU-35 Reporte de empleados no afiliados al IESS

HISTORIA USUARIO	
Número: HU-35	Nombre Historia de Usuario: Reporte de quienes no están afiliados al IESS.
Usuario: Administrador	Sprint Asignado: 10
Fecha Inicio: 16/06/17	Fecha Fin: 16/06/17
Descripción: Como administrador necesito visualizar el reporte de empleados que no estén afiliados al IESS Para tener un control de dichos empleados y realizar las acciones correspondientes.	
Pruebas de aceptación: <ul style="list-style-type: none">– Verificar que se visualice el reporte de empleados no afiliados al IESS en formato PDF	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-35 se define en la siguiente tabla.

Tabla 265-3 Actividades de HU-35

FECHA	ACTIVIDAD	TIEMPO
16/06/17- 16/06/17	Tarea 1: Desarrollo del reporte de quienes no están afiliados al IESS. <ul style="list-style-type: none">• Crear el método de consulta obtenerEmpleadoIESS en la clase NegocioEmpleado• Crear el método obtenerEmpleadoNoIESS en la clase ControladorEmpleado• Crear el método initEmpleadosNoIESS en la clase ControladorEmpleado• Crear el método PDFEmpleadosNoIESS la clase ControladorEmpleado• Diseño de reporte en IReport	8 horas

Realizado por: Erika R. 2017

Tabla 266-3 Tarea de Ingeniería para la HU-35

TAREAS DE INGENIERÍA	
SPRINT: 09	Número de Tarea: 01
Nombre de Historia de Usuario: Reporte de quienes no están afiliados al IESS.	
Nombre de la Tarea: Implementación del reporte de quienes no están afiliados al IESS.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 16/06/17	Fecha Fin: 16/06/17
Descripción <ul style="list-style-type: none"> • Crear el método de consulta obtenerEmpleadoIESS en la clase NegocioEmpleado • Crear el método obtenerEmpleadoNoIESS en la clase ControladorEmpleado • Crear el método initEmpleadosNoIESS en la clase ControladorEmpleado • Crear el método PDFEmpleadosNoIESS la clase ControladorEmpleado • Diseño de reporte en IReport 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Verificar que se descarga y visualice el reporte de quienes no están afiliados al IESS. 	

Realizado por: Erika R. 2017

Tabla 267-3 Prueba de aceptación 1 para tarea de de ingenieria 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación del reporte de quienes no están afiliados al IESS.
Nombre de la Prueba: Verificar que se descarga y visualice el reporte de quienes no están afiliados al IESS.	
Responsable: Erika Reina	Fecha: 16/06/17
Descripción: Se descargará el reporte de empelados que no están afiliados al IESS en un archivo formato PDF el mismo que se abrirá en el navegador para su visualización respectiva.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber empleados registrados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador • Clic en Reportes/Empleados no afiliados al IESS. • Clic en el reporte PDF descargado • Visualizar la lista de empelados no afiliados al IESS. 	

<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se visualiza el reporte PDF en el navegador respectivo.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 36

Tabla 268-3 HU-36 Reporte de asistencias por departamento

HISTORIA USUARIO	
Número: HU-36	Nombre Historia de Usuario: Reporte de asistencia por departamento.
Usuario: Administrador	Sprint Asignado: 10
Fecha Inicio: 19/06/17	Fecha Fin: 19/06/17
Descripción: Como administrador necesito visualizar el reporte de asistencia de los empleados por departamento para tener un control del cumplimiento a cabalidad de su jornada de trabajo.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Verificar que se visualice el reporte de asistencia por departamento en formato PDF 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-36 se define en la siguiente tabla.

Tabla 269-3 Actividades de HU-36

FECHA	ACTIVIDAD	TIEMPO
19/06/17- 19/06/17	<p>Tarea 1: Desarrollo del reporte de asistencia por departamento.</p> <ul style="list-style-type: none"> • Crear el método de consulta obtenerAsistenciaDepar en la clase NegocioEmpleado • Crear el método obtenerAsistenciaDepar en la clase ControladorEmpleado • Crear el método initAsistenciaDepar en la clase ControladorEmpleado • Crear el método PDFAsistenciaDepar la clase ControladorEmpleado 	8 horas

	<ul style="list-style-type: none"> • Diseño de reporte en IReport 	
--	--	--

Realizado por: Erika R. 2017

Tabla 270-3 Tarea de Ingeniería para la HU-36

TAREAS DE INGENIERÍA	
SPRINT: 09	Número de Tarea: 01
Nombre de Historia de Usuario: Reporte de asistencia por departamento.	
Nombre de la Tarea: Implementación del reporte de asistencia por departamento.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 19/06/17	Fecha Fin: 19/06/17
Descripción <ul style="list-style-type: none"> • Crear el método de consulta obtenerAsistenciaDepar en la clase NegocioEmpleado • Crear el método obtenerAsistenciaDepar en la clase ControladorEmpleado • Crear el método initAsistenciaDepar en la clase ControladorEmpleado • Crear el método PDFAsistenciaDepar la clase ControladorEmpleado • Diseño de reporte en IReport 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Verificar que se descargue y visualice el reporte de asistencia por departamento. 	

Realizado por: Erika R. 2017

Tabla 271-3 Prueba de aceptación 1 para tarea de de ingenieria 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación del reporte de asistencia por departamento.
Nombre de la Prueba: Verificar que se descargue y visualice el reporte de asistencia por departamento.	
Responsable: Erika Reina	Fecha: 19/06/17
Descripción: Se descargará el reporte de las asistencias de empelados por departamento en un archivo formato PDF el mismo que se abrirá en el navegador para su visualización respectiva.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber empleados registrados. 	
Pasos de Ejecución <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador • Clic en Reportes/Asistencia por departamento. • Clic en el reporte PDF descargado 	

<ul style="list-style-type: none"> • Visualizar la lista de empedados y sus asistencias organizados por departamentos.
Resultado Esperado: <ul style="list-style-type: none"> • Se visualiza el reporte PDF en el navegador respectivo.
Evaluación de la Prueba: <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 37

Tabla 272-3 HU-37 Reporte de vacaciones gozadas, por gozar y saldos

HISTORIA USUARIO	
Número: HU-37	Nombre Historia de Usuario: Reporte de vacaciones gozadas, por gozar y saldos.
Usuario: Administrador	Sprint Asignado: 10
Fecha Inicio: 20/06/17	Fecha Fin: 20/06/17
Descripción: Como administrador necesito visualizar el reporte de vacaciones gozadas, por gozar y saldos para tener un control de los días que aún tiene disponibles vacaciones, así como las que ya ha tomado y realizar un respectivo análisis.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Verificar que se visualice el reporte de vacaciones gozadas, por gozar y saldos en formato PDF 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-37 se define en la siguiente tabla.

Tabla 273-3 Actividades de HU-37

FECHA	ACTIVIDAD	TIEMPO
20/06/17- 20/06/17	Tarea 1: Desarrollo del reporte de vacaciones gozadas, por gozar y saldos. <ul style="list-style-type: none"> • Crear el método de consulta obtenerVacaciones en la clase NegocioEmpleado • Crear el método obtenerVacaciones en la clase ControladorEmpleado 	8 horas

	<ul style="list-style-type: none"> • Crear el método initVacaciones en la clase ControladorEmpleado • Crear el método PDFVacaciones la clase ControladorEmpleado • Diseño de reporte en IReport 	
--	--	--

Realizado por: Erika R. 2017

Tabla 274-3 Tarea de Ingeniería para la HU-37

TAREAS DE INGENIERÍA	
SPRINT: 09	Número de Tarea: 01
Nombre de Historia de Usuario: Reporte de vacaciones gozadas, por gozar y saldos.	
Nombre de la Tarea: Implementación del reporte de vacaciones gozadas, por gozar y saldos.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 20/06/17	Fecha Fin: 20/06/17
Descripción <ul style="list-style-type: none"> • Crear el método de consulta obtenerVacaciones en la clase NegocioEmpleado • Crear el método obtenerVacaciones en la clase ControladorEmpleado • Crear el método initVacaciones en la clase ControladorEmpleado • Crear el método PDFVacaciones la clase ControladorEmpleado • Diseño de reporte en IReport 	
Pruebas de Aceptación: <ul style="list-style-type: none"> • Verificar que se descarga y visualice el reporte de vacaciones gozadas, por gozar y saldos. 	

Realizado por: Erika R. 2017

Tabla 275-3 Prueba de aceptación 1 para tarea de de ingenieria 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación del reporte de vacaciones gozadas, por gozar y saldos.
Nombre de la Prueba: Verificar que se descarga y visualice el reporte de vacaciones gozadas, por gozar y saldos.	
Responsable: Erika Reina	Fecha: 20/06/17
Descripción: Se descargará el reporte de vacaciones gozadas, por gozar y saldos en un archivo formato PDF el mismo que se abrirá en el navegador para su visualización respectiva.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Debe haber conexión con la base de Datos. • Deben haber empleados registrados. 	

<p>Pasos de Ejecución</p> <ul style="list-style-type: none"> • Ingresar al sistema con las credenciales del Administrador • Clic en Reportes/Vacaciones. • Clic en el reporte PDF descargado • Visualizar la lista de empelados y sus vacaciones.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se visualiza el reporte PDF en el navegador respectivo.
<p>Evaluación de la Prueba:</p> <ul style="list-style-type: none"> • Satisfactoria

Realizado por: Erika R. 2017

Historia de usuario 38

Tabla 276-3 HU-38 Reporte de vacaciones acumuladas

HISTORIA USUARIO	
Número: HU-38	Nombre Historia de Usuario: Reporte de vacaciones acumuladas.
Usuario: Administrador	Sprint Asignado: 10
Fecha Inicio: 21/07/17	Fecha Fin: 21/07/17
Descripción: Como administrador necesito visualizar el reporte de vacaciones acumuladas para tener un control de los días que totales de vacaciones que dispone un empleado.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> – Verificar que se visualice el reporte de vacaciones acumuladas en formato PDF 	

Realizado por: Erika R. 2017

La tarea de ingeniería realizada en la HU-38 se define en la siguiente tabla.

Tabla 277-3 Actividades de HU-38

FECHA	ACTIVIDAD	TIEMPO
21/07/17- 21/07/17	<p>Tarea 1: Desarrollo del reporte de vacaciones acumuladas.</p> <ul style="list-style-type: none"> • Crear el método de consulta obtenerVacacionesAcumuladas en la clase NegocioEmpleado • Crear el método obtenerVacacionesAcumuladas en la clase 	8 horas

	<p>ControladorEmpleado</p> <ul style="list-style-type: none"> • Crear el método initVacacionesAcumuladas en la clase ControladorEmpleado • Crear el método PDFVacacionesAcumuladas la clase ControladorEmpleado • Diseño de reporte en IReport 	
--	---	--

Realizado por: Erika R. 2017

Tabla 278-3 Tarea de Ingeniería para la HU-38

TAREAS DE INGENIERÍA	
SPRINT: 09	Número de Tarea: 01
Nombre de Historia de Usuario: Reporte de vacaciones acumuladas.	
Nombre de la Tarea: Implementación del reporte de vacaciones acumuladas.	
Tipo de Tarea: Desarrollo	Programador Responsable: Erika Reina.
Fecha de Inicio: 21/07/17	Fecha Fin: 21/07/17
<p>Descripción</p> <ul style="list-style-type: none"> • Crear el método de consulta obtenerVacacionesAcumuladas en la clase NegocioEmpleado • Crear el método obtenerVacacionesAcumuladas en la clase ControladorEmpleado • Crear el método initVacacionesAcumuladas en la clase ControladorEmpleado • Crear el método PDFVacacionesAcumuladas la clase ControladorEmpleado • Diseño de reporte en IReport 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Verificar que se descarga y visualice el reporte de vacaciones acumuladas. 	

Realizado por: Erika R. 2017

Tabla 279-3 Prueba de aceptación 1 para tarea de de ingenieria 1

PRUEBAS DE ACEPTACIÓN	
Código: 1.1	Nombre de la Tarea de Ingeniería: Implementación del reporte de vacaciones acumuladas.
Nombre de la Prueba: Verificar que se descarga y visualice el reporte de vacaciones acumuladas.	
Responsable: Erika Reina	Fecha: 21/07/17
Descripción: Se descargará el reporte de vacaciones acumuladas en un archivo formato PDF el mismo que se abrirá en el navegador para su visualización respectiva.	
Condiciones de Ejecución:	

- Debe haber conexión con la base de Datos.
- Deben haber empleados registrados.

Pasos de Ejecución

- Ingresar al sistema con las credenciales del Administrador
- Clic en Reportes/Vacaciones acumuladas.
- Clic en el reporte PDF descargado
- Visualizar la lista de empelados y sus vacaciones acumuladas.

Resultado Esperado:

- Se visualiza el reporte PDF en el navegador respectivo.

Evaluación de la Prueba:

- Satisfactoria

Realizado por: Erika R. 2017