



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

**ANÁLISIS DE RENDIMIENTO ENTRE POSTGRESQL Y SQL SERVER
USANDO HAMMERDB Y MANAGE ENGINE APLICADO AL SISTEMA
ACADÉMICO DE CONDUESPOCH**

TESIS DE GRADO

Previa la obtención del título de

INGENIERO EN SISTEMAS INFORMÁTICOS

Presentado por:

MARIO FERNANDO ANDRADE ARRIETA

JOHNNY DARÍO PARRA GARCÍA

RIOBAMBA – ECUADOR

2014

AGRADECIMIENTO

Nuestros más sinceros agradecimientos a la Escuela Superior Politécnica de Chimborazo y sus docentes quienes nos brindaron sus valiosos conocimientos durante nuestra etapa de estudios para alcanzar una educación de calidad.

A cada uno de nuestros familiares quienes supieron apoyarnos de manera incondicional y nos ayudaron a culminar de manera exitosa la presente investigación.

A nuestro director de tesis el Ing. Paúl Paguay y asesora la Ing. Germania Veloz quienes con sus conocimientos y experiencias aportaron significativamente al desarrollo de esta Tesis, induciéndonos a la investigación, a la excelencia y al trabajo de calidad que es lo que propone la educación del siglo XXI, guiándonos y brindándonos todo su valioso tiempo en el desarrollo de la misma.

DEDICATORIA

Esta trabajo de investigación lo dedico en primera instancia a Dios por las bendiciones y fuerzas recibidas día a día lo que me ha motivado para poder proyectarme de manera positiva en cada una de mis metas para llegar a ser un profesional útil para la sociedad, de manera muy especial a mi padre José quien en su momento de vida supo brindarme todo su apoyo y ahora a pesar de su ausencia siempre me sirvió como fuente de inspiración para poder cumplir cada uno de mis objetivos, de igual manera a mi querida madre María que constantemente confió en mis capacidades y ha sido mi pilar fundamental en los buenos y malos momentos de mi vida brindándome su cariño, apoyo y comprensión. Así mismo a mis queridos hermanos Milton, Oswaldo, Andrea y Darwin quienes fueron un puntal de inspiración, demostrando en todo momento su confianza depositada en mí.

Johnny

La presente Tesis, fruto del esfuerzo constante y de un sueño mediato dibujado en la mente de un estudiante sencillo pero con grandes proyectos ofrezco en primer lugar al Todo Poderoso y a la Madre del Cielo quienes me han fortalecido en los momentos álgidos y de desesperanza, proyectándome la energía necesaria para continuar en los caminos a veces tortuosos, además la dedico con profundo respeto y amor, a aquellos seres que creyeron en mi compromiso de cada día y no vacilaron jamás en apoyarme en mis anhelos de forjarme un futuro de esperanzas y realidades, ellos son mi padre ausente que desde el infinito sideral iluminó mi camino para cumplir mis metas, supongo que compartirá mi alegría y se sentirá orgulloso de mi trabajo; a mi madre y a mi hermana que lucharon incansablemente para lograr un título profesional que será el baluarte de mi vida laboral y forjará un destino promisorio, sirviendo honrosamente a nuestra sociedad.

Mario

FIRMAS RESPONSABLES Y NOTAS

NOMBRE	FIRMAS	FECHA
ING. GONZALO NICOLAY SAMANIEGO DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
DR. JULIO SANTILLÁN CASTILLO DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS	_____	_____
ING. PAÚL PAGUAY DIRECTOR DE TESIS	_____	_____
ING. GERMANIA VELOZ MIEMBRO DEL TRIBUNAL	_____	_____
DIRECTOR DEL CENTRO DE DOCUMENTACIÓN	_____	_____

NOTA: _____

RESPONSABILIDAD DE LOS AUTORES

Nosotros, Mario Fernando Andrade Arrieta y Johnny Darío Parra García, somos responsables de las ideas, doctrinas y los resultados expuestos en esta tesis, y el patrimonio intelectual de la Tesis de Grado pertenece a la Escuela Superior Politécnica de Chimborazo

Mario Fernando Andrade Arrieta

Johnny Darío Parra García

ABREVIATURAS

T-SQL = Transact SQL

SGBD = Sistema de Gestión de Base de Batos

MVCC= Multiversion Concurrency Control

DQL = Doctrine Query Language

WAL = Write Ahead Log

SI = Sistema Informático

BSD = Berkeley Software Distribution

SSL = Secure Sockets Layer

JSON = JavaScript Object Notation

SAN = Storage Area Network

GAM & SGAM = (Global Allocation Map & Shared Global Allocation Map)

PFS = Page Free Space

IAM = Index Allocation Map

BCM = Bulk Changed Map

DCM = Differential Change Map

API = Application Programming Interface

OS = Operating System

CLR = Common Language Runtime

TIC = Tecnologías de la Información y Comunicación

ACID = Atomicidad, Consistencia, Insolación, durabilidad

ÍNDICE GENERAL

AGRADECIMIENTO

DEDICATORIA

ABREVIATURAS

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

INTRODUCCIÓN

CAPÍTULO I

1.	MARCO REFERENCIAL.....	14
1.1	ANTECEDENTES	14
1.2	JUSTIFICACIÓN	16
1.2.1	Justificación teórica	16
1.2.2	Justificación práctica	18
1.3	OBJETIVOS	20
1.3.1	Objetivo General	20
1.3.2	Objetivos Específicos.....	20
1.4	HIPÓTESIS	21

CAPÍTULO II

2.	MARCO TEÓRICO.....	22
2.1	SISTEMA DE BASE DE DATOS	22
2.1.1	Introducción.....	22
2.1.2	Objetivos de los sistemas de base de datos.....	23
2.2	BASE DE DATOS POSTGRESQL	25
2.2.1	Introducción.....	25
2.2.2	Historia	25
2.2.3	Arquitectura de PostgreSQL	26
2.2.4	Ventajas y Limitaciones.....	29
2.2.5	Orientación a objetos en PostgreSQL.....	30
2.2.6	Herramientas de administración.....	31
2.2.7	Características de la versión 9.2.4	32
2.2.8	Instalación	35
2.2.9	Licencia	35
2.3	BASE DE DATOS SQL SERVER	36

2.3.1	Introducción a SQL Server	36
2.3.2	Historia de SQL-Server	37
2.3.3	Arquitectura de SQL Server	38
2.3.4	Características generales.....	40
2.3.5	Ventajas y Limitaciones.....	42
2.3.6	Instalación e implementación.....	43
2.3.7	Conectarse al motor de Base de datos	43
2.3.8	Conectarse a Management Studio.....	44
2.3.9	Autorizar conexiones adicionales.....	46
2.3.10	Crear un inicio de sesión con autenticación de Windows	46
2.3.11	Conectarse desde otro equipo	47
2.3.12	Habilitar protocolos.....	47
2.3.13	Habilitar conexiones TCP/IP desde otro equipo	47
2.3.14	Configurar un puerto fijo.....	48
2.3.15	Configurar SQL Server para escuchar en un puerto específico	49
2.3.16	Conectarse al motor de base de datos desde otro equipo	50
2.3.17	Administración de la Base de Datos	51
2.3.18	Procesamiento de instrucciones SQL.....	53
2.3.19	Creación de base de datos en SQL Server	54
2.4	EL RENDIMIENTO EN LOS SGBD	55
CAPÍTULO III		
3.	ESTUDIO DE HERRAMIENTAS Y DEFINICIÓN DE MÉTRICAS.....	57
3.1	Investigación de herramientas para medir el rendimiento de los SGBD	57
3.1.1	Herramientas de pruebas de carga.....	58
3.1.2	Herramientas de monitoreo.....	66
3.2	Selección de las herramientas	72
3.3	Definición de las métricas	73
3.3.1	Consumo de recursos	73
3.3.2	Tiempo de respuesta.....	75
3.3.3	Número de órdenes por minuto.....	75
3.3.4	Tiempo del aseguramiento de la integridad de datos	75
3.4	Determinación de los escenarios de prueba.....	77
3.4.1	Esquemas de Base de datos para las pruebas	79
3.5	Diseño de las pruebas.....	84
3.5.1	Prueba 1 - Consumo de recursos	85
3.5.2	Prueba 2 - tiempos de respuesta	86
3.5.3	Prueba 3 - Número de órdenes por minuto.....	88

3.5.4	Prueba 4 – Tiempo del aseguramiento de la integridad de datos	90
3.5.5	Métodos para la evaluación de resultados.....	92
CAPÍTULO IV		
4.	ANÁLISIS COMPARATIVO ENTRE LOS SGBD POSTGRESQL Y SQL SERVER	96
4.1	Ejecución de las pruebas y tabulación de resultados	96
4.1.1	Prueba 1 - consumo de recursos	96
4.1.2	Prueba 2 - Tiempo de respuesta.....	104
4.1.3	Prueba 3 - Número de órdenes por minuto.....	111
4.1.4	Prueba 4 – Tiempo del aseguramiento de la integridad de datos	114
4.2	Análisis de los datos obtenidos	116
4.3	Comprobación de la hipótesis.....	118
4.3.1	Determinación de la población y muestra	118
4.3.2	Planteamiento de la hipótesis	119
4.3.3	Comprobación de la hipótesis.....	120
CAPÍTULO V		
5.	DESARROLLO DEL SISTEMA ACADÉMICO APLICANDO EL RESULTADO DEL ANÁLISIS A LOS SGBD.	121
5.1	INFORMACION GENERAL DE LA EMPRESA	121
5.1.1	Antecedentes	121
5.1.2	Visión.....	122
5.1.3	Misión	122
5.2	BENEFICIOS DEL SISTEMA.....	122
5.2.1	Beneficios tangibles	122
5.2.2	Beneficios intangibles.....	122
5.3	FACTIBILIDAD	123
5.3.1	Factibilidad Técnica.....	123
5.3.2	Factibilidad Económica	123
5.3.3	Factibilidad Operativa.....	124
5.4	ESTIMACIONES	124
5.5	ANÁLISIS DE RIESGOS.....	125
5.6	PLANIFICACIÓN GENERAL DEL PROYECTO	125
5.7	ARQUITECTURA DEL SISTEMA	126
5.7.1	Diagrama de despliegue	127
5.7.2	Diagrama de componentes	128
5.8	METODOLOGÍA SCRUM	128
5.8.1	Características	129
5.8.2	Ventajas.....	129

5.8.3	Roles de SCRUM	130
5.8.4	Fases.....	131
5.9	METODOLOGÍA SCRUM APLICADA AL DESARROLLO DEL SISTEMA ACADÉMICO DE “CONDUESPOCH”.....	133
	REQUERIMIENTOS FUNCIONALES.....	133
	PRODUCT BACKLOG	135
	SPRINT BACKLOG.....	142
	SISTEMA ACADÉMICO CONDUESPOCH BURN UP.....	150
	CODIFICACIÓN DEL SISTEMA	150
	DISEÑO DE INTERFACES DE USUARIO PARA EL SISTEMA ACADEMICO CONDUESPOCH.....	151
	ESQUEMA DE BASE DE DATOS DEL SISTEMA	154
	CONCLUSIONES	
	RECOMENDACIONES	
	RESUMEN	
	SUMMARY	
	GLOSARIO	
	ANEXOS	
	BIBLIOGRAFÍA	

ÍNDICE DE TABLAS

Tabla I.I: Herramientas para medir el rendimiento de los SGBD	17
Tabla I.II: Módulos del sistema académico para CONDUESPOCH	19
Tabla II.III: Limitaciones de PostgreSQL	29
Tabla II.IV: Enfoque de PostgreSQL para el concepto de clase y objeto	30
Tabla II.V: Capacidad de cálculo de SQL Server	41
Tabla II.VI: Alta disponibilidad de SQL Server	41
Tabla II.VII: Herramientas Administrativas de SQL Server	42
Tabla III.VIII: Estudios y White papers que han usado Hammerdb	61
Tabla III.IX: Empresas que utilizan Manage Engine	69
Tabla III.X: Herramientas de pruebas de carga en BD	72
Tabla III.XI: Herramientas para monitorear el consumo de recursos.....	73
Tabla III.XII: Tablas y cantidad de registros usados en las pruebas de CR y NOPM.....	79
Tabla III.XIII: Prueba 1 – Consumo de recursos	85
Tabla III.XIV: Prueba 2 – Tiempo de respuesta	88
Tabla III.XV: Prueba 3 – Número de órdenes por minuto	90
Tabla III.XVI: Prueba 4 - Tiempo del aseguramiento de la integridad de datos.....	91
Tabla III.XVII: Valores referenciales para evaluar el rendimiento de los SGBD	92
Tabla III.XVIII: Rango de valores para la evaluación de los recursos CPU y RAM	93
Tabla III.XIX: Valores referenciales para evaluar el tiempo de respuesta	94
Tabla IV.XX: Pruebas para consumo de recursos	96
Tabla IV.XXI: Resultados de la prueba uso de CPU.....	97
Tabla IV.XXII: Resultados de la prueba uso de memoria RAM	99
Tabla IV.XXIII: Resultados de la prueba uso de RED.....	100
Tabla IV.XXIV: Resultado de los porcentajes ponderados de la prueba consumo de recursos	102
Tabla IV.XXV: Resultado de los porcentajes de la prueba consumo de recursos.....	103
Tabla IV.XXVI: Escenarios de la prueba tiempo de respuesta	104
Tabla IV.XXVII: Resultado total de la prueba TDR, inserción de registros	105
Tabla IV.XXVIII: Resultado total de la prueba TDR, consulta de registros	108
Tabla IV.XXIX: Resultado de los porcentajes de la prueba tiempo de respuesta	110
Tabla IV.XXX: Resultado de los porcentajes de la prueba tiempo de respuesta	110
Tabla IV.XXXI: Resultados de la prueba 3 (NOPM).....	112
Tabla IV.XXXII: Resultados de la prueba de Integridad.....	115
Tabla IV.XXXIII: Resultado general de las pruebas	117
Tabla V.XXXIV: Resultado general de las pruebas.....	124
Tabla V.XXXV: Product Backlog	136
Tabla V.XXXVI: Sprint Backlog	143
Tabla V.XXXVII: Sprint 1	148
Tabla V.XXXVIII: Historia de usuario 1 del sprint 1	148
Tabla V.XXXIX: Tarea 1 de la historia de usuario 1	149

ÍNDICE DE FIGURAS

Figura I.1: Módulos propuestos para el desarrollo del sistema	19
Figura II.2: Componentes de un sistema de base de datos	22
Figura II.3: Componentes de PostgreSQL.....	27
Figura II.4: ecosistema de la Plataforma de Información Cloud Ready.....	37
Figura II.5: Evolución de la plataforma de datos de Microsoft.....	38
Figura III.6: Herramienta SQL Query Stress.....	58
Figura III.7: Herramienta Hammerdb	61
Figura III.8: Herramienta Advanced Query Tool (AQT)	64
Figura III.9: Herramienta Database Stress Testing Tool.....	66
Figura III.10: Herramienta Manage Engine.....	69
Figura III.11: Herramienta Task Manager	70
Figura III.12: Performance Advisor for SQL Server	71
Figura III.13: Ejemplo integridad referencial	77
Figura III.14: Esquema utilizado para la realización de las pruebas	78
Figura III.15: Esquema de Base de datos para las pruebas.....	80
Figura III.16: Sistema de cómputo de órdenes	81
Figura III.17: Parámetros de configuración para TPC-C.....	81
Figura III.18: Configuración de usuarios virtuales para TPC-C	82
Figura III.19: Esquema 2 de base de datos para las pruebas	84
Figura III.20: Interfaz pruebas de tiempo de respuesta	87
Figura III.21: Interfaz de salida de las pruebas en TPC-C.....	89
Figura IV.22: Porcentajes de la prueba uso de CPU	97
Figura IV.23: Promedio general prueba uso de CPU	98
Figura IV.24: Porcentajes de la prueba uso de memoria RAM	99
Figura IV.25: Promedio general de la prueba uso de RAM	100
Figura IV.26: Cantidad de paquetes recibidos en la prueba uso de RED	101
Figura IV.27: Promedio general de la prueba uso de RED.....	101
Figura IV.28: Resultado general de la prueba consumo de recursos.....	103
Figura IV.29: Resultados de la prueba 2 (TDR), inserción de registros	106
Figura IV.30: Resultado general de la prueba de inserción de registros.....	107
Figura IV.31: Resultados de la prueba 2 (TDR), consulta de registros	108
Figura IV.32: Resultado parcial de la prueba 2 Consulta de registros sobre el 100' %	109
Figura IV.33: Resultado total de la prueba tiempo de respuesta.....	111
Figura IV.34: Resultados de la prueba 3 (NOPM)	113
Figura IV.35: Resultado total de la prueba número de órdenes por minuto sobre el 100' % ...	113
Figura IV.36: Resultado prueba tiempo aseguramiento de la integridad de datos.....	115
Figura IV.37: Resultado total de la prueba Tiempo del aseguramiento de la integridad de datos	116
Figura IV.38: Resultados general de las pruebas	117
Figura IV.39: Resultado general de la investigación.....	118
Figura V.40: Arquitectura Modelo Vista Controlador	126
Figura V.41: Diagrama de despliegue.....	127
Figura V.42: Diagrama de despliegue.....	128
Figura V.43: Fases metodología SCRUM.....	132
Figura V.44: Fases metodología SCRUM.....	149
Figura V.45: Sistema Académico CONDUESPOCH burn up	150
Figura V.46: Vista del sistema: Perfil Secretaria.....	152
Figura V.47: Reporte de acta de calificaciones.	153

INTRODUCCIÓN

La presente investigación se enfoca a estudiar y determinar cuál SGBD ofrece mejor rendimiento para el desarrollo de aplicaciones web, estableciendo parámetros de comparación, los mismos que serán analizados y probados en diferentes tipos de pruebas.

En el Capítulo I Marco referencial, se detalla los antecedentes, la justificación de la investigación, los objetivos a cumplirse y la hipótesis planteada, la misma que será comprobada al final de la investigación.

En el Capítulo II Marco teórico, comprende una investigación de los SGBD SQL Server y PostgreSQL, donde se detallan sus historias, características, funciones, arquitectura, etc. De manera similar una breve introducción a los sistemas de base de datos.

En el capítulo III Estudio de Herramientas y métricas, se describe el estudio de las herramientas para realizar las pruebas de carga y las herramientas de monitoreo de recursos. Así como también se definirá cada una de las métricas a evaluar, se determinará los escenarios de pruebas y los métodos para la evaluación de resultados de la presente investigación.

El Capítulo IV Análisis comparativo entre los SGBD PostgreSQL y SQL Server, presenta los resultados correspondientes del estudio previamente realizado y la determinación del mejor SGBD.

En el Capítulo V se detalla el desarrollo de cada uno de los módulos del Sistema académico para la Escuela de conducción CONDUESPOCH con el SGBG de mejor rendimiento.

CAPÍTULO I

1. MARCO REFERENCIAL

En este capítulo se tratará sobre el marco referencial, en el cual se encuentra de manera general los antecedentes, la justificación del proyecto de tesis, los objetivos a alcanzar con el desarrollo de la misma y el planteamiento de la hipótesis que se desea demostrar.

1.1 ANTECEDENTES

Las bases de datos son la forma más habitual para el almacenamiento estructurado de datos tanto en pequeñas aplicaciones de escritorio pasando por aplicaciones móviles y las más grandes aplicaciones multiusuario debido a su capacidad para mantener la integridad de información, ofrecer mayor disponibilidad, acceso más rápido y sencillo para los usuarios finales y la facilidad para compartir los datos por el conjunto de usuarios.

Es necesario analizar todo el proceso a lo largo de cualquier proyecto que implique trabajar con base de datos dado que la tecnología asociada y conceptos son heterogéneos para asegurar el éxito.

Hoy por hoy las herramientas existentes en el mercado para gestionar base de datos son variadas y están orientadas a diferentes áreas de acuerdo a la necesidad como por ejemplo: Oracle, DB2, Microsoft SQL Server, Sybase, MySQL, Cloudscape/Derby, PostgreSQL, entre otros.

Debido a que no existe ninguna investigación que se enfoque en el análisis de rendimiento de las herramientas para gestión de base de datos entre PostgreSQL 9.2.4 y SQL Server Express Edition 2012 el presente estudio se enfocará en la comparación de los Sistemas de gestión de Base de Datos antes mencionados.

PostgreSQL es un poderoso sistema de base de datos, objeto-relacional de base de datos, con más de 15 años de desarrollo activo, el mismo que posee una arquitectura muy sólida con lo cual se ha ganado una alta reputación de fiabilidad e integridad de datos. Es altamente compatible con varios sistemas operativos del mercado como Windows, Linux y UNIX (Tru64, Solaris, AIX, BSD, HP-UX, SGI IRIX, Mac OS X,). Provee un alto grado de compatibilidad con ACID, posee más de SQL: 2008 tipos de datos, entre los que se destaca TIMESTAMP, INTEGER, CHAR, VARCHAR, DATE, INTERVAL E INTEGER. Contiene soporte para vistas, claves principales y foráneas, uniones, disparadores y procedimientos almacenados **[45]**.

Microsoft SQL Server es un Sistema Gestor de Base de datos relacionales, está basado en el lenguaje Transact-SQL y es capaz de procesar grandes cantidades de información de muchos usuarios simultáneamente y de forma muy rápida **[42]**.

Las organizaciones pueden utilizar SQL Server para proteger de manera eficiente, desbloquear, y escalar el poder de sus datos a través de aplicaciones de escritorio, dispositivos móviles, centro de datos y a partir de la versión 2012 en una nube privada o pública. Se proporciona a las organizaciones con un rendimiento de misión crítica y disponibilidad, así como el potencial para desbloquear ideas innovadoras con el descubrimiento de datos generalizada a través de la organización **[42]**.

Actualmente la Escuela De Capacitación Para Conductores Profesionales “CONDUESPOCH” lleva el control de su información de forma semi-automatizada, mediante el uso de hojas

electrónicas de Excel para el almacenamiento de información de carácter administrativo-académico y por otra parte se utilizan también hojas de papel que son llenadas a mano, ambas están definidas como instrumentos que proporcionan información a los funcionarios, docentes y estudiantes con respecto a calificaciones, carga horaria, pagos pendientes de estudiantes, etc.

El creciente volumen de información que se genera y gestiona en la institución actualmente, hace que dicha información no este localizada en el momento en que se necesita; esto provoca un consumo adicional e innecesario de tiempo y recursos.

Ante la falta de una herramienta que mantenga la integridad en los datos, evite la duplicación de procesos, que establezca la responsabilidad en cada uno de sus usuarios de acuerdo a sus funciones y mejore la eficiencia de los procesos que actualmente son poco eficientes "CONDUESPOCH" requiere una herramienta que cumpla con características como: usabilidad, mantenibilidad, portabilidad y que resuelva las problemáticas antes mencionadas.

1.2 JUSTIFICACIÓN

1.2.1 Justificación teórica

El factor de rendimiento en los sistemas de gestión de base de datos hoy en día juega un papel importante al momento de elegir uno de ellos, con el fin de que las bases de datos puedan funcionar con total normalidad, al mismo tiempo que se debe tener en cuenta los recursos de hardware y software necesarios para que puedan responder de manera eficiente a la demanda de las distintas transacciones que día a día tenga que atender el sistema de gestión de base de datos

En una determinada empresa, el manejo de los problemas de rendimiento es algo que compete a toda la empresa, de manera especial a los administradores de las base de datos quienes deberán resolver problemas de rendimiento causado por distintos componentes de las TIC's como: servidores, red, sistemas operativos, es por ello que se debe contar con una adecuada infraestructura de las TIC's que permita responder eficazmente cuando surgen problemas de rendimiento.

Es así que con el análisis que se va realizar se determinará cuál SGBD consume menos recursos de hardware mediante la utilización de herramientas de software, las cuales proporcionarán resultados estadísticos sobre los recursos consumidos.

La **tabla I.I** describe algunas herramientas para medir el rendimiento de los SGBD, a continuación se describe las funciones cada una de ellas

Tabla I.I: Herramientas para medir el rendimiento de los SGBD

Fuente: Johnny Parra, Mario Andrade

HERRAMIENTAS	DESCRIPCIÓN
<ul style="list-style-type: none">• Hammerdb• Database Stress TestingTool• SQLQueryStress	Realizan miles transacciones de manera directa y automática con los Sistemas de gestión de bases de datos. Su objetivo es generar gran cantidad de carga en el servidor de base de datos.
<ul style="list-style-type: none">• Manage Engine• Windows Task Manager• SQLSentry	Mediante estas herramientas se podrá monitorear de forma constante el uso de los recursos en el sistema, luego de efectuar las pruebas de carga.

Mediante el uso de las herramientas antes mencionadas se va medir el consumo de recursos del sistema como:

- Uso de memoria RAM
- Uso de CPU
- Uso de Red

De igual manera se tomará en cuenta otros parámetros como:

- Tiempos de respuesta al efectuar las transacciones (inserciones y consultas)
- Número de órdenes por minuto realizadas en un intervalo de tiempo
- Tiempo del aseguramiento de la integridad de datos.

En primer lugar se utilizará algunas de las herramientas antes mencionadas, las cuales interactuarán de manera directa con el servidor de base de datos, efectuando carga de trabajo mediante el uso de miles transacciones que serán realizadas por varios usuarios (virtuales) a la vez, adicionalmente se creará un prototipo de aplicación web de tal manera que desde la misma se pueda realizar un conjunto de transacciones; para lo cual se tomará en cuenta varios parámetros y posteriormente generar gráficos estadísticos sobre los mismos.

De esta manera esta investigación permitirá conocer cuál de los SGBD SQL Server o PostgreSQL brinda un mejor rendimiento a la hora de realizar varios tipos de transacciones.

1.2.2 Justificación práctica

La implementación del sistema académico para la empresa pública CONDUESPOCH, se enfoca a resolver inconvenientes en los distintos procesos de preinscripción, matriculación, asignación de notas, generación de reportes, entre otros.

Las ventajas de su implementación son las siguientes:

- Integridad de los datos,
- Menor consumo de tiempo en cada uno de los procesos
- Niveles de seguridad
- Portabilidad
- Usabilidad

La **figura I. 1** con los respectivos módulos que contendrá el sistema académico para CONDUESPOCH.



Figura I.1: Módulos propuestos para el desarrollo del sistema

Fuente: Johnny Parra, Mario Andrade

En base a la **figura I.1** se procederá a explicar de manera general cada uno de ellos en la **tabla I.II**; para el uso del sistema académico se tendrá 5 roles: docente, inspector, secretaria, administrador, pedagógico.

Tabla I.II: Módulos del sistema académico para CONDUESPOCH

Fuente: Johnny Parra, Mario Andrade

MÓDULO	FUNCIÓN	USUARIOS
Inscripción	Se registra los alumnos con su información básica y se valida requisitos preestablecidos para el efecto	Secretaria
Matriculación	<ul style="list-style-type: none"> • Validar Inscripción • Registro de matrícula en el sistema • Asignación de materias al estudiante 	Secretaria
Notas	• Registro de notas y asistencia de estudiantes	Docente / Inspector / Secretaria
	• Consulta de notas	Docente / Secretaria
Reportes	<ul style="list-style-type: none"> • Notas de estudiantes • Alumnos aprobados / reprobados / suspensos 	Docente
Administración general	<ul style="list-style-type: none"> • Ingresar, modificar <ul style="list-style-type: none"> ○ Materias ○ Docentes ○ Usuarios del sistema ○ Etc. 	Administrador

Para el desarrollo de la aplicación web se plantea la utilización de una metodología ágil como lo es SCRUM.

SCRUM: Se caracteriza por proporcionar flexibilidad y ser ágil al momento de gestionar el desarrollo de software. La misma que tiene como finalidad construir primero los requerimientos de mayor importancia y valor para el cliente, basándose en principios como la auto-gestión, adaptación y la inspección continua [3].

1.3 OBJETIVOS

1.3.1 Objetivo General

- Realizar un análisis de rendimiento entre PostgreSQL y SQL Server mediante el uso de las herramientas Hammerdb y Manage Engine aplicado al desarrollo del sistema académico para la empresa pública CONDUESPOCH, permitiendo obtener la herramienta de SGBD de mejor rendimiento

1.3.2 Objetivos Específicos

- Investigar las características de los Sistemas de Gestión de Base de Datos PostgreSQL y SQL Server.
- Realizar un análisis de rendimiento entre los Sistemas de Gestión de Base de Datos PostgreSQL y SQL Server.
- Determinar los parámetros y herramientas para evaluar el rendimiento de Sistemas de Gestión de Base de Datos.
- Realizar un prototipo de aplicación web como mecanismo que ayudará a medir el rendimiento entre PostgreSQL y SQL Server.
- Evaluar los resultados obtenidos del estudio previo de los SGBD PostgreSQL y SQL Server.
- Escoger el Sistema de Gestión de Base de Datos de mejor rendimiento en aplicaciones web.

- Desarrollar un Sistema para la Escuela De Capacitación Para Conductores Profesionales “CONDUESPOCH”.

1.4 HIPÓTESIS

El SGBD PostgreSQL ofrece un mejor rendimiento frente al SGBD SQL Server en la implementación de aplicaciones web.

CAPÍTULO II

2. MARCO TEÓRICO

2.1 SISTEMA DE BASE DE DATOS

2.1.1 Introducción

Un sistema de base de datos es un sistema automatizado que brinda información para el manejo de base de datos mediante paquetes de software llamados Sistema de gestión de base datos [40].

Entre los componentes principales para el desarrollo de un sistema de base de datos se encuentran: software, hardware y datos [40].

El componente software se divide en dos partes que son: SGBD y la Aplicación de base de datos que es un programa que a través de interfaces gráficas a permite al usuario visualizar, recuperar y actualizar la información [40].

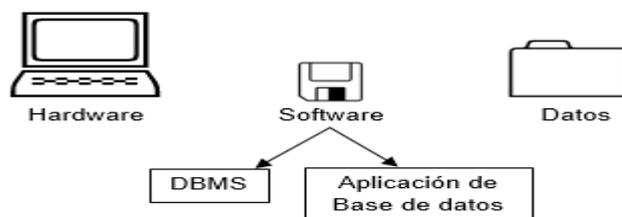


Figura II.2: Componentes de un sistema de base de datos

Fuente: Texto básico: Base de datos I: Rodríguez Flores Ivonne Elizabeth

La satisfacción o insatisfacción del usuario al usar un sistema de base de datos tiene relación directa entre la solicitud realizada y tiempo de respuesta, a mayor tiempo de respuesta el valor del sistema decrece **[40]**.

2.1.2 Objetivos de los sistemas de base de datos

Un sistema de base de datos trata de alcanzar a cumplir los siguientes objetivos:

- Integridad de datos
- Seguridad de los datos
- Independencia de los datos
- Minimizar la redundancia de los datos
- Control de concurrencia y simultaneidad

Integridad de datos

La integridad es mantener correctos los datos en la base de datos mediante medidas de seguridad que permiten el acceso a múltiples usuarios **[40]**.

Las fuentes potenciales de error que pueden introducir errores en la base de datos son:

- Defectos de software.- los errores en el software pueden existir en programas de aplicación y pueden generarse debido a errores en la lógica, el responsable de que no ocurran estos errores y de solucionarlos es el programador **[40]**.
- Errores humanos.- se refieren a errores en el ingreso de datos no válidos, errores de digitación **[40]**.
- Actualización incompleta.- es probable que se produzcan estos errores si se está ejecutando actualizaciones de la base de datos y ante una falla eléctrica por ejemplo la actualización no se completa, entonces se tiene inconsistencia en la base de datos **[40]**.
- Falla de equipo.- los golpes, el polvo y humedad en el disco duro pueden comprometer su integridad, entonces los datos pueden destruirse físicamente **[40]**.

Seguridad de los datos

La seguridad se refiere a la protección de la base de datos contra acceso y modificaciones que no están autorizadas, no todos los datos y no todos los usuarios son creados de la misma manera ni con los mismos privilegios, un fallo en la seguridad hace a la base de datos vulnerable a intromisiones [40].

Independencia de los datos

Existen dos formas de independencia de datos que son:

- Independencia física de datos: la independencia física está asociada al hardware, existe independencia física cuando los cambios en la organización física de la BD no afectan a la aplicación (programas, usuarios) [40].
- Independencia lógica de datos: la independencia lógica está asociada al software, existe independencia lógica cuando los cambios en el esquema lógico no afectan a la aplicación [40].

Minimizar la redundancia de los datos

Los mismos datos que son almacenados varias veces en la base de datos producen problemas como:

- Desperdicio en el espacio de almacenamiento.
- Posibilidad de información inconsistente.

Control de concurrencia y simultaneidad

Es necesario mecanismos de control de concurrencia que aseguren que ninguna transacción concurrente pueda interferir con el resto de operaciones generadas por varios usuarios tratando de acceder a la misma base de datos [40].

2.2 BASE DE DATOS POSTGRESQL

2.2.1 Introducción

PostgreSQL es un poderoso sistema de base de datos, objeto-relacional de base de datos, con más de 15 años de desarrollo activo, el mismo que posee una arquitectura muy sólida con lo cual se ha ganado una alta reputación de fiabilidad e integridad de datos [45].

Altamente compatible con varios sistemas operativos del mercado como Windows, Linux y UNIX (Tru64, Solaris, AIX, BSD, HP-UX, SGI IRIX, Mac OS X,). Provee un alto grado de compatibilidad con ACID, posee más de SQL: 2008 tipos de datos, entre los que se destaca TIMESTAMP, INTEGER, CHAR, VARCHAR, DATE, INTERVAL E INTEGER. Contiene soporte para vistas, claves principales y foráneas, uniones, disparadores y procedimientos almacenados [45].

2.2.2 Historia

Al igual que con muchos otros proyectos clave de código abierto, tuvo sus inicios en la Universidad de California en Berkeley (UCB) [45].

PostgreSQL tuvo sus inicios en la Universidad de California en Berkeley (UCB), en la cual originalmente fue llamado Postgres, fue creado por Michael Stonebraker un profesor de ciencias de la computación, el mismo que posteriormente alcanzó el grado de director de tecnología de la corporación Informix. Su desarrollo ahora basado en la teoría clásica de RSGBD continuó desde 1977 hasta 1985 con el nombre de Ingres pero esta vez era de propiedad de Computer Associates [45].

Postgres en cambio fue desarrollado entre 1986-1994, el cual esta vez se basaba en conceptos de bases de datos tales como la exploración de las tecnologías de objetos relacionales, donde se introdujo nuevas normas, procedimientos, tipos extensibles con índices y conceptos objeto-relacional. Posteriormente Postgres fue adquirido por Informix para luego IBM hacerse dueño de sí mismo [45].

En 1995 dos estudiantes Andrew Yu y Jolly Chen quienes contaban con estudios de PhD, dentro de los laboratorios de Stonebraker, crearon Postquel, el cual contenía un subconjunto extendido de SQL, por lo cual cambió su nombre a Postgres95 [45].

En 1996 un grupo de desarrolladores vieron la promesa del sistema y continuaron su desarrollo fuera de la Berkeley, orientados a una nueva vida en el mundo del código abierto. El fruto de su trabajo era una nueva base de datos que ganó una reputación de estabilidad sólida, con muchas nuevas características y mejoras, el sistema de base de datos tomó el nuevo y último nombre: PostgreSQL y su versión inicial fue la 6.0, durante los próximos años se crearon nuevas mejoras y características como:

- **Control de Concurrencia Multiversión (MVCC).** que permite a los lectores a seguir leyendo datos consistentes durante la actividad de escritura y permite realizar copias de seguridad en línea (en caliente) [45].
- **Características importantes de SQL.-** Se realizaron muchas mejoras de SQL incluyendo subconsultas, valores predeterminados, las restricciones, claves primarias, claves foráneas, identificadores entre comillas, cohesión de tipos de datos string, la conversión de tipos y entrada de enteros binarios y hexadecimales entre otros [45].
- **Mejora de tipos integrados.** Se han añadido nuevos tipos nativos incluyendo una amplia gama de tipos de date/time y los tipos geométricos adicionales [45].

2.2.3 Arquitectura de PostgreSQL

PostgreSQL utiliza un modelo cliente/servidor. Una sesión PostgreSQL consiste de la siguiente colaboración de procesos (programas).

- **Un proceso servidor:** Administra los archivos de las base de datos, acepta conexiones a la base de datos desde las aplicaciones cliente y realiza acciones en la base de datos de parte de los clientes. Este proceso es llamado Postgres.

- **La aplicación cliente (frontend):** es quien realiza las operaciones hacia la base de datos. Las aplicaciones cliente pueden ser de diversa naturaleza: pueden ser herramientas de orientadas al texto, aplicaciones gráficas, un servidor web que acceda a la base de datos para desplegar páginas web o una herramienta de administración de base de datos especializada.
- Como es típico en las aplicaciones cliente-servidor, el cliente y servidor pueden estar en diferentes equipos, en el caso que su comunicación sea mediante una red TCP/IP.

Su arquitectura utiliza multiprocesos en lugar de *multihilos* para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando [21].

La **figura II. 3** ilustra de manera general los componentes más importantes en un sistema PostgreSQL [21].

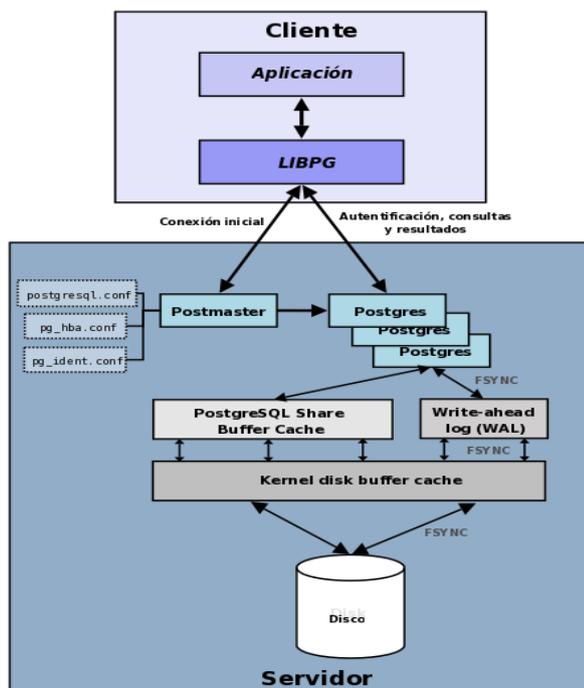


Figura II.3: Componentes de PostgreSQL

Fuente: http://www.PostgreSQL.org.es/sobre_PostgreSQL

Donde un proceso servidor (postmaster) y una serie de aplicaciones cliente que realizan solicitudes de acciones contra la base de datos a su proceso servidor. Por cada una de estas aplicaciones cliente, el proceso postmaster crea un proceso postgres.

- **Aplicación cliente:** Esta es la aplicación cliente que utiliza PostgreSQL como administrador de bases de datos. La conexión puede ocurrir vía TCP/IP o sockets locales [21].

- **Demonio postmaster:** Este es el proceso principal de PostgreSQL. Es el escuchar por un puerto/socket por conexiones entrantes de clientes. También es el encargado de crear los procesos hijos que se encargaran de autenticar estas peticiones, gestionar las consultas y mandar los resultados a las aplicaciones clientes [21].

- **Ficheros de configuración:** Los 3 ficheros principales de configuración son:
 - **Postgresql.conf:** fichero de configuración principal, contiene la asignación a los parámetros que configuran el funcionamiento del servidor [21].
 - **pg_hba.conf:** fichero de configuración de la autenticación de los clientes y usuarios y del acceso a las bases de datos del cluster [21].
 - **pg_ident.conf:** fichero accesorio al anterior, determina como se realiza la autenticación *ident* que contiene la correspondencia entre usuarios del Sistema Operativo y de PostgreSQL [21].

- **Procesos hijos postgres:** Procesos hijos que se encargan de autenticar a los clientes, de gestionar las consultas y mandar los resultados a las aplicaciones clientes [21].

- **PostgreSQL share buffer cache:** Memoria compartida usada por PostgreSQL para almacenar datos en caché y gestionar los recursos entre procesos backend [21].

- **Write-Ahead Log (WAL):** Componente del sistema encargado de asegurar la integridad de los datos (recuperación de tipo REDO) [21].

- **Kernel disk buffer cache:** Caché de disco del sistema operativo [21].

- **Disco:** Disco físico donde se almacenan los datos y toda la información necesaria para que PostgreSQL funcione [21].

2.2.4 Ventajas y Limitaciones

Ventajas [45]

- Ideal para modelos de negocio de gran escala
- Flexibilidad para realizar investigaciones y pruebas sin necesidad de incluir costes de licencia adicionales.
- Flexibilidad y estabilidad
- El código fuente está disponible para todos sin costo alguno
- Es multiplataforma
- Diseñado para entornos de alto volumen, utiliza una estrategia de almacenamiento de datos de varias filas denominado MVCC.
- Existe una cantidad de Interfaces gráficas de usuario de base de datos y herramientas de administración

Limitaciones

La **tabla II.III** proporciona algunas limitaciones del SGBD PostgreSQL en su versión 9.2.4

Tabla II.III: Limitaciones de PostgreSQL

Fuente: http://www.PostgreSQL.org.es/sobre_PostgreSQL

LÍMITE	VALOR
Máximo tamaño base de datos	Ilimitado – (con una dependencia del sistema de almacenamiento) [21].
Máximo tamaño de tabla	32 TB
Máximo tamaño de fila	1.6 TB
Máximo tamaño de campo	1 GB
Máximo número de filas por tabla	Ilimitado
Máximo número de columnas por tabla	250 a 1600 (con dependencia del tipo)
Máximo número de índices por tabla	Ilimitado

2.2.5 Orientación a objetos en PostgreSQL

Lo que tienen a favor las bases de datos objeto-relacionales es que permiten realizar una migración gradual de sistemas relacionales a los orientados a objetos y, en algunos casos, coexisten los dos tipos de aplicaciones durante algún tiempo [21].

El inconveniente de este enfoque es que no es muy fácil conseguir la compatibilidad de dos modelos de datos diferentes como son la orientación a objetos y el modelo relacional. Por lo cual es necesario nivelar de alguna manera los conceptos de los dos modelos sin que entren en problemas [21].

Uno de los conceptos primordiales en la orientación a objetos es el concepto de clase. Existen dos enfoques para asociar el concepto de clase con el modelo relacional [21]:

- 1-er enfoque: las clases definen tipos de tablas
- 2.º enfoque: las clases definen tipos de columnas

Debido a que en el modelo relacional las columnas están determinadas por tipos de datos, lo más natural es hacer relacionarse las columnas con las clases [21].

La **tabla II.IV** muestra el enfoque orientado a objetos de PostgreSQL

Tabla II.IV: Enfoque de PostgreSQL para el concepto de clase y objeto

Fuente: <http://www.dataprix.com/bases-datos-PostgreSQL>

	PRIMER ENFOQUE	SEGUNDO ENFOQUE
Los objetos son	Valores	Tuplas
Las clases son	Dominios	Tablas

PostgreSQL implementa los objetos como tuplas y las clases como tablas. De manera similar es posible definir nuevos tipos de datos mediante los mecanismos de extensión [21].

Dado que las tablas son clases, cabe la posibilidad de definirse como herencia de otras. Las tablas derivadas son polimorfas y heredan todos los atributos (columnas) de la tabla padre (incluida su clave primaria). Si no se manipulan con precaución, las tablas polimorfas pueden acarrear a errores de integridad al duplicar claves primarias [21].

PostgreSQL soporta algunas extensiones del lenguaje SQL para crear y gestionar este tipo de tablas.

2.2.6 Herramientas de administración

A continuación se detallan algunas herramientas más utilizadas en el mercado, las mismas que son de código libre.

❖ PgAdmin III

Es la herramienta Open Source con un soporte completo para UNICODE, edición rápida de consultas y datos multihilo, a la vez que brinda soporte para todos los tipos de objetos de PostgreSQL [44].

Para su administración dispone de una herramienta con interfaz gráfica para el trabajo con SQL, un editor de código de procedimientos y funciones. Es compatible con plataformas como: MS Windows, GNU/Linux, FreeBSD, Mac OS X, OpenBSD, Solaris [44].

• PhpPgAdmin

Hace uso de un navegador web y brinda soporte para consultas ad-hoc, además para la navegación y modificación de la mayoría de los tipos de objetos de las bases de datos PostgreSQL [44].

Una vez conocidas algunas herramientas de administración gratuitas, de igual manera se presenta otras de tipo comercial.

- **Lightning Admin for PostgreSQL**

Altamente compatible con Windows y en otras plataformas bajo el uso de WINE, esta herramienta dispone de una interfaz gráfica tipo MDI [44].

- **BDOne**

Permite la administración de base de datos como: PostgreSQL, Oracle, MSSQL, Sybase, DB2, SAPDB, Interbase/Firbird y MySQL. Únicamente compatible con Windows [44].

2.2.7 Características de la versión 9.2.4

INTRODUCCIÓN

PostgreSQL 9.2.4 se libera el 10 de septiembre de 2012 con soporte nativo para JSON, índices de cobertura, mejoras de replicación y de desempeño, y muchas otras características [46].

- **Rendimiento y escalabilidad mejorados**

Las mejoras en la escalabilidad vertical de PostgreSQL optimizan la utilización eficaz de los recursos de hardware en servidores más potentes. Los avances en gestión de bloqueos, eficiencia en escritura, acceso de únicamente índices y de igual manera en operaciones de bajo nivel permiten que PostgreSQL tenga la posibilidad de manejar cargas de trabajo de volumen aún mal altas. En cuanto a números, esto significa [46]:

- Permite realizar hasta 350.000 consultas de lectura por segundo
- Consultas en almacenes de datos con acceso sólo de índices (2-20 veces más rápido)
- Hasta 14.000 escrituras de datos por segundo (5 veces más rápido)
- Hasta un 30% de disminución de utilización de energía por la CPU

De manera similar la adición de replicación en cascada permite a los usuarios que puedan procesar lotes más grandes mediante servidores escalados horizontalmente.

- **Soporte flexible para desarrolladores**

La versión 9.2.4 brinda una alta flexibilidad para aplicaciones de misión crítica, mediante la inclusión de soporte para tipos de datos intervalo y JSON, lo que permite a los desarrolladores usar PostgreSQL de maneras totalmente nuevas [46].

Los tipos de datos de intervalo (range types) permiten a los desarrolladores crear excelentes aplicaciones de agendamiento, científicas y financieras, lo que ningún otro motor SQL implementa esta característica en la actualidad, lo cual permite una gestión inteligente de intervalos de tiempo y números [46].

Con PostgreSQL 9.2.4, el resultado de una consulta se puede devolver en forma de datos JSON. Combinado con las nuevas extensiones de programación PL/V8 ¹para Javascript y PL/Cooffee, y el almacén de llave-valor HStore opcional, los usuarios ahora pueden utilizar PostgreSQL como base de datos de documentos “NoSQL” manteniendo a la vez la fiabilidad, flexibilidad y rendimiento de PostgreSQL [46].

El soporte nativo para JSON en PostgreSQL proporciona un mecanismo eficaz para la creación y el almacenamiento de documentos para APIs web.

INFORMACIÓN GENERAL

Esta versión se ha centrado en gran medida en las mejoras de rendimiento. El trabajo también se ha enfocado en el área de soporte de la replicación. Las principales mejoras contemplan:

- Permite realizar consultas con el fin de recuperar únicamente los datos de los índices, evitando el acceso montón (*exploraciones de índice solamente*)

¹ Lenguaje procedural

- Permite al planificador generar planes personalizados para los valores con parámetros específicos, incluso cuando se utilicen declaraciones preparadas
- Mejorar la capacidad del planificador al utilizar bucles anidados con exploraciones de unión de índices.
- Permitir la transmisión de replicación de esclavos para enviar datos a otros esclavos (replicación en cascada)
 - Permite que un servidor en espera/reserva pueda aceptar conexiones de replicación y transmitir registros WAL a otro en espera, actuando como un relé /relay. Esto hace que se reduzca el número de conexiones directas con el maestro y también para reducir al mínimo los gastos generales entre sitios de banda ancha
- Permite realizar copias de seguridad (pg_basebackup) de base de servidores en espera
 - pg_basebackup hace una copia binaria de los archivos de un clúster de base de datos, mientras que se asegura que el sistema esté disponible y fuera del modo de copia de seguridad automáticamente
- Añade una herramienta (pg_receivexlog) para registrar los cambios del archivo WAL cuando se escriben
 - pg_receivexlog transmite el registro de transacciones en tiempo real, el que esté siendo generado en el servidor, y no espera a sus segmentos para completarlo; así como lo hace archive_command
- Añade (SP-GiST) (Espacio con particiones GiST) método de acceso a índices
 - SP-GiST está destinado a permitir el desarrollo de tipos de datos personalizados con los métodos de acceso adecuados, por un experto en el dominio del tipo de datos, en lugar de un experto en la base de datos.

- Añade soporte para tipos de datos de *range*
 - *range*: son útiles porque representan muchos valores de los elementos en un solo valor de alcance, y porque conceptos como rangos superpuestos pueden expresarse con claridad
- Añade el tipo de datos *JSON*
 - *json* puede ser utilizado para almacenar tipos de datos JSON (JavaScript Object Notation). Estos datos también se pueden almacenar como texto, pero el tipo de datos *json* tiene la ventaja de la comprobación de que cada valor almacenado es un valor JSON válido.
- Añade una opción (*security_barrier*) para visitas
- Permite cadenas de conexión *libpq* para tener formato de una URI
- Añadir un modo de procesamiento de una sola fila para que *libpq* mejore el manejo de resultados de grandes conjuntos.

2.2.8 Instalación

Se va presentar una guía de instalación del Sistema de Gestión de Base de Datos (SGBD) PostgreSQL en su versión **9.2.4**. La misma que se puede encontrar en el **Anexo I**

2.2.9 Licencia

PostgreSQL hace uso de la licencia PostgreSQL, algo similar a la licencia BSD², que tiene como único requerimiento que mantenga en su código fuente su información de copyright y licencia. Esta licencia es certificada por OSI, a la vez es reconocida por ser muy flexible y amigable para los negocios, debido a que no restringe el uso de PostgreSQL con aplicaciones comerciales y privadas **[45]**.

² Licencia *Berkeley Software Distribution*

2.3 BASE DE DATOS SQL SERVER

2.3.1 Introducción a SQL Server

Microsoft SQL Server es un Sistema Gestor de Base de datos relacionales, está basado en el lenguaje Transact-SQL y es capaz de procesar grandes cantidades de información de muchos usuarios simultáneamente y de forma muy rápida [42].

SQL Server 2012 es una plataforma de Microsoft información preparada para la nube. Las organizaciones pueden utilizar SQL Server 2012 para proteger de manera eficiente, desbloquear, y escalar el poder de sus datos a través de aplicaciones de escritorio, dispositivos móviles, centro de datos, y ya sea una nube privada o pública. Basándose en el éxito del SQL Server 2008 versión R2, SQL Server 2012 ha hecho un fuerte impacto en las organizaciones en todo el mundo con sus capacidades significativas. Se proporciona a las organizaciones con un rendimiento de misión crítica y disponibilidad, así como el potencial para desbloquear ideas innovadoras con el descubrimiento de datos generalizada a través de la organización. Finalmente, SQL Server 2012 ofrece una variedad de soluciones híbridas que puede elegir. Por ejemplo, una organización puede desarrollar e implementar aplicaciones y soluciones de base de datos en entornos no virtualizados, en appliances³, y en On-premises⁴ de nubes privadas o fuera de las instalaciones en nubes públicas. Por otra parte, estas soluciones pueden integrar fácilmente uno con el otro, ofreciendo una solución híbrida totalmente integrada [42].

La **Figura II.4** ilustra el ecosistema de la Plataforma de Información Cloud Ready.

³ Dispositivo diseñado para un uso o función particular

⁴ Método tradicional de la instalación y personalización de software en los ordenadores propios de los clientes que residen en el interior de su propio centro de datos.

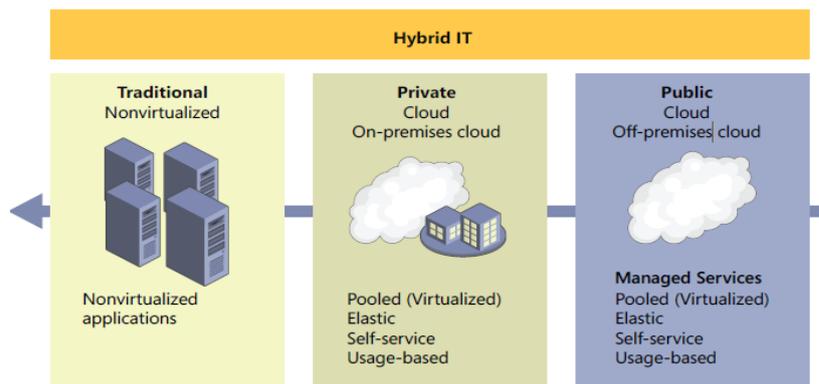


Figura II.4: ecosistema de la Plataforma de Información Cloud Ready

Fuente: Introducing Microsoft® SQL Server 2012; Ross Mistry and Stacia Misner

Hoy en día las organizaciones necesitan confianza, rentabilidad y una plataforma escalable de base de datos que ofrece la confianza de misión crítica, ideas innovadoras y flexibles. Las organizaciones siempre se deben ajustar porque las tendencias nuevas e importantes son regularmente cambiantes. Microsoft ha realizado importantes inversiones en el producto SQL Server 2012 en su conjunto, sin embargo, las principales nuevas características y capacidades innovadoras que deben interesar a los administradores de bases de datos (DBAs) son: disponibilidad, escalabilidad, rendimiento, seguridad y administración [42].

2.3.2 Historia de SQL-Server

Según lo investigado SQL Server empezó en el año de 1989 a partir SQL Server 1.0 que se ejecutaba bajo el sistemas operativo (OS/2), en el año de 1993 surge SQL Server 4.21 para WinNT, luego en el año de 1995 sale SQL Server 6.0 para el SO Windows 95, en el año 1996 hace su aparición SQL Server 6.5, SQL Server 7.0 surge en el año 1998, al año siguiente 1999 Microsoft lanzó SQL Server 7.0 OLAP Tools (On-Line Analytical Processing), la versión 8.0 también llamada SQL Server 2000 aparece en el año 2000, SQL Server 2000 64-bit Edition surge en el año 2003, en el 2005 es lanzada la versión SQL Server 2005 o versión 9.0, luego en el 2008 es lanzada la versión 10.0 o SQL Server 2008, la siguiente versión de este motor nace en el año 2010 como SQL Server 2008 R2, dos años más adelante aparece SQL Server 2012 y finalmente SQL Server 2014 que es la última versión de SQL Server [10] [18].

La **figura II.5** representa la evolución de la plataforma de datos de Microsoft

The Evolution of Microsoft Data Platform

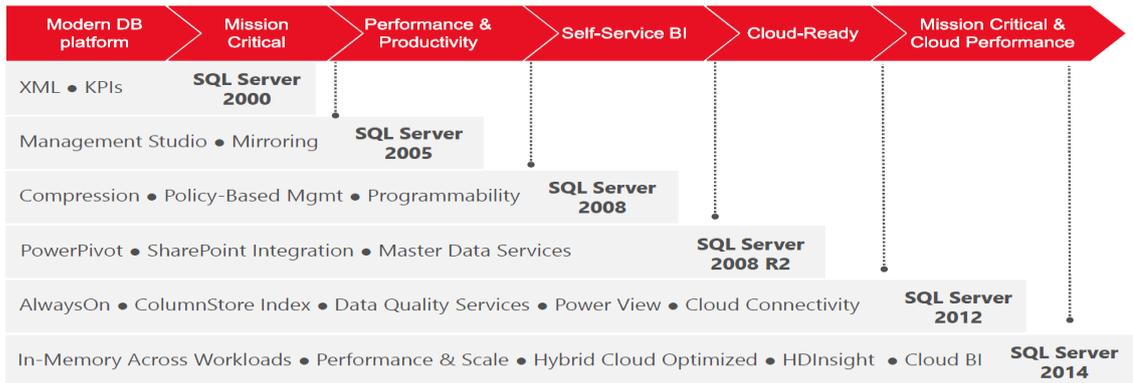


Figura II.5: Evolución de la plataforma de datos de Microsoft

Fuente: <http://drsalbertspijkers.blogspot.com/2014/03/sql-server-2014-ctp-and-ata-intensive.html>

2.3.3 Arquitectura de SQL Server

El concepto de arquitectura en SQL Server es un muy profundo, al ser un tópico demasiado extenso se lo va exponer de forma breve y siendo lo más concreto posible; los componentes más importantes de SQL Server son:

1. Relational Engine
2. Storage Engine
3. SQL OS

El **Relational Engine** es también llamado procesador de consultas, incluye los componentes de SQL Server que determinan exactamente que consulta necesita realizar y la mejor manera de hacerlo, se encarga de gestionar la ejecución de las consultas y ya que solicita los datos desde el motor de almacenamiento procesa los resultados devueltos [37].

Estas son las tareas del Relational Engine:

1. Query Processing
2. Memory Management
3. Thread and Task Management
4. Buffer Management

5. Distributed Query Processing

El **Storage Engine** es responsable del almacenamiento y recuperación de los datos sobre el sistema de almacenamiento (SAN, Disco, etc.), para entender más se enfocan los siguientes conceptos [37].

Cuando se habla de cualquier base de datos en SQL Server, existen 2 tipos de archivos que son creados a nivel de disco que son el Data file y el Data Log. El Data file almacena físicamente los datos en páginas de datos (data pages). Los Log files que son conocidos también como write ahead logs (WAL), estos son usados para transacciones de almacenamiento ejecutadas sobre la base de datos [37].

El Data File almacena datos en forma de Data Page (8KB) y estas páginas de datos son organizadas lógicamente en extensiones (extents) [37].

Los Extents son unidades lógicas en la base de datos, son combinaciones de 8 data pages, por ejemplo 64KB forman una extent. Las extensiones (extents) pueden ser de 2 tipos, Mixed y Uniform. Las extensiones Mixed tienen diferentes tipos de pages como: index, system, datas, etc. Por otro lado las extensiones Uniform están dedicadas solamente a un tipo (object) [37].

Con respecto a las Pages como debemos saber qué tipo de páginas de datos pueden almacenarse en SQL Server, a continuación se mencionan algunas de ellas:

- **Data Page:** tienen la data insertada por el usuario pero no la data que es de tipo text, ntext, nvarchar(max), varchar(max), varbinary(max), image and xml data [37].
- **Index:** almacena las entradas index [37].
- **Text/Image:** almacena LOB (Large Object data) Como text, ntext, varchar (max), nvarchar (max), varbinary (max), image and xml data [37].
- **GAM & SGAM** (Global Allocation Map & Shared Global Allocation Map): son usados para guardar información relacionado a la localización de extents [37].
- **PFS** (Page Free Space): información relacionada a la localización de la página y al espacio no usado disponible en las páginas (pages) [37].

- **IAM** (Index Allocation Map): información relativa a las extensiones que son usadas por una table o index por unidad de asignación [37].
- **BCM** (Bulk Changed Map): mantiene la información acerca de las extensiones cambiadas en una operación masiva (Bulk) [37].
- **DCM** (Differential Change Map): esta es la información de las extensiones que se han modificado desde la última instrucción de backup de la base de datos como por unidad de asignación [37].

El Log File también conocido como write ahead log (WAL). Almacena las modificaciones de la base de datos (DML, DDL) [37].

Información suficiente es registrada (logged) para poder:

- Revertir las transacciones si así lo solicita.
- Recuperar la base de datos en caso de falla.
- Write Ahead Logging es usado para crear entradas de registro.

Los registros de transacciones son escritas en orden cronológico de forma circular.

La política de truncamiento de registros se basa en el modelo de recuperación.

El **SQL OS** se encuentra entre el equipo host (Windows OS) y SQL Server. Todas las actividades realizadas en el motor de base de datos son atendidos por SQL OS. Es un sistema operativo altamente configurable con potente API (interfaz de programación de aplicaciones), que permite localidad automática y el paralelismo avanzado. SQL OS ofrece varios servicios del sistema operativo, tales como ofertas de gestión de memoria con pool de buffer, Log Buffer y la detección de estancamiento mediante el bloqueo y la estructura de bloqueo. Otros servicios incluyen el manejo de excepciones, de alojamiento para los componentes externos como Common Language Runtime, CLR, etc [37].

2.3.4 Características generales

A continuación se exponen algunas de las características más sobresalientes en las diferentes versiones de SQL Server 2012.

La **tabla II.V** expone las capacidades de cálculo de las diferentes versiones de SQL Server 2012.

Tabla II.V: Capacidad de cálculo de SQL Server

Fuente: <http://redondo.j.wordpress.com/2012/02/20/ediciones-de-ms-sql-server-2012/>

Nombre de la característica	Enterprise	Bussiness Intelligence	Standard	Web	Express con Advanced Services	Express con herramientas	Express
Capacidad máxima de cálculo utilizada por una instancia única	Sistema operativo máximo	Limitada a menos de 4 sockets o 16 núcleos	Limitada a menos de 4 sockets o 16 núcleos	Limitada a menos de 4 sockets o 16 núcleos	Limitada a menos de 1 sockets o 4 núcleos	Limitada a menos de 1 sockets o 4 núcleos	Limitada a menos de 1 sockets o 4 núcleos
Memoria máxima usada	Sistema operativo máximo	64 GB	64 GB	64 GB	1 GB	1 GB	1 GB
Tamaño máximo de la base de datos relacional	524 TB	524 TB	524 TB	524 TB	10 GB	10 GB	10 GB

La **tabla II.VI** expone las capacidades de disponibilidad de las diferentes versiones de SQL Server 2012.

Tabla II.VI: Alta disponibilidad de SQL Server

Fuente: <http://redondo.j.wordpress.com/2012/02/20/ediciones-de-ms-sql-server-2012/>

Nombre de la característica	Enterprise	Bussiness Intelligence	Standard	Web	Express con Advanced Services	Express con herramientas	Express
Compatibilidad con Server Core	Si	Si	Si	Si	Si	Si	Si
Trasvase de registros	Si	Si	Si	Si			
Creación de reflejo de la base de datos	Si	Si(sólo seguridad completa)	Si(sólo seguridad completa)	Sólo testigo	Sólo testigo	Sólo testigo	Sólo testigo

La **tabla II.VII** expone las herramientas administrativas de las diferentes versiones de SQL Server 2012.

Tabla II.VII: Herramientas Administrativas de SQL Server

Fuente: <http://redondo.j.wordpress.com/2012/02/20/ediciones-de-ms-sql-server-2012/>

Nombre de la característica	Enterprise	Bussiness Intelligence	Standard	Web	Express con Advanced Services	Express con herramientas	Express
Objetos de administración de SQL (SMO)	Si	Si	Si	Si	Si	Si	Si
Administración de configuración de SQL	Si	Si	Si	Si	Si	Si	Si
SQL CMD (herramienta del símbolo del sistema)	Si	Si	Si	Si	Si	Si	Si
SQL Server Management Studio	Si	Si	Si	Si	Si	Si	

2.3.5 Ventajas y Limitaciones

Ventajas

Para la versión SQL Server Express Edition entre las ventajas se tiene:

- SQL Express Server es gratis.
- Su instalación es sencilla.
- Puede funcionar en la mayoría del hardware actual.
- Todas las versiones de SQL Server 2012, a partir de la libre Express Edition hasta la Enterprise Edition pueden tener un máximo de 50 instancias.
- SQL Manager Configuración, SQL Command Prompt y SQL Server Management Studio son soportados.

Limitaciones

A continuación se detalla las limitaciones del SGBD SQL Server Express Edition versión 2012 [8].

- El volumen máximo de la base de datos está limitado a 10Gb [8].
- Límite de 1 socket o 4 núcleos para el procesamiento de la base de datos [8].
- El máximo de memoria que usará la instancia será de 1gb [8].
- Puede activarse como base de datos espejo (mirroring), aunque solo como testigo [8].
- Puede ser suscriptora en un servicio de replicación [8].
- Incluye también los servicios de Búsqueda de Texto (fulltext search) [8].
- Permite un limitado almacenamiento FILESTREAM [8].

2.3.6 Instalación e implementación.

Se va a presentar una guía de instalación del Sistema de Gestión de Base de Datos (SGBD) SQL Server en su versión 2012 Express Edition La misma que se puede encontrar en el **Anexo I**.

2.3.7 Conectarse al motor de Base de datos

El motor de base de datos de SQL Server incluye varias herramientas. En esta sección se describen las primeras herramientas que se necesitará y se ofrece ayuda para seleccionar la herramienta adecuada para la tarea. Se puede obtener acceso a todas las herramientas desde el menú Inicio. No se instalan las mismas herramientas, como SQL Server Management Studio, de forma predeterminada. Deben ser seleccionadas como parte de los componentes de cliente durante la instalación [23].

Herramientas básicas

- SQL Server Management Studio es la herramienta principal para administrar el motor de base de datos y escribir código de Transact-SQL. No se incluye en SQL Server Express,

pero está disponible como descarga independiente en el Centro de descarga de Microsoft [23].

- El administrador de configuración de SQL Server se instala con SQL Server y las herramientas cliente. Permite habilitar protocolos de servidor, configurar opciones de protocolo, como los puertos TCP, configurar servicios de servidor para que se inicien automáticamente y configurar equipos cliente para que se conecten de la forma preferida. Esta herramienta configura los elementos de conectividad más avanzados, pero no habilita las características [23].

Para iniciar SQL Server Management Studio

- En el menú Inicio, seleccionar Todos los programas, Microsoft SQL Server 2012 y, a continuación, hacer clic en SQL Server Management Studio [23].

Para iniciar el Administrador de configuración de SQL Server

- En el menú Inicio, seleccionar Todos los programas, seleccionar Microsoft SQL Server 2012, seleccionar Herramientas de configuración y, a continuación, hacer clic en Administrador de configuración de SQL Server [23].

2.3.8 Conectarse a Management Studio

Para conectarse al Motor de base de datos desde herramientas que se ejecutan en el mismo equipo es necesario conocer el nombre de la instancia y si se conecta como miembro del grupo Administradores del equipo. Los siguientes procedimientos deben realizarse en el mismo equipo en el que se hospeda SQL Server [23].

Para determinar el nombre de la instancia de motor de base de datos

- Iniciar una sesión en Windows como miembro del grupo Administradores y abrir Management Studio [23].
- En el cuadro de diálogo Conectar con el servidor, hacer clic en Cancelar [23].

- Si servidores registrados no aparece, en el menú Ver, hacer clic en Servidores registrados **[23]**.
- Con el motor de base de datos seleccionado en la barra de herramientas Servidores registrados, expandir Motor de base de datos, hacer clic con el botón secundario en Grupos de servidores locales, seleccionar Tareas y, a continuación, hacer clic en Registrar servidores locales. Se muestran todas las instancias de Motor de base de datos instaladas en el equipo. La instancia predeterminada no tiene nombre y aparece como el nombre del equipo. Una instancia con nombre aparece como el nombre del equipo seguido de una barra inversa (\) y del nombre de la instancia. En SQL Server Express, la instancia se denomina <computer_name>sqlexpress a no ser que se haya cambiado el nombre durante la instalación **[23]**.

Para comprobar que el motor de base de datos está en ejecución

- En Servidores registrados, si el nombre de la instancia de SQL Server tiene un punto verde con una flecha blanca junto al nombre, Motor de base de datos está en ejecución y no es necesario realizar ninguna otra acción **[23]**.
- Si el nombre de la instancia de SQL Server tiene un punto rojo con un cuadrado blanco junto al nombre, Motor de base de datos se encuentra detenido. Hacer clic con el botón secundario en el nombre de Motor de base de datos, hacer clic en Control de servicios y, a continuación, hacer clic en Iniciar. Después de un cuadro de diálogo de confirmación, Motor de base de datos debería iniciarse y el color del punto debería cambiar a verde con una flecha blanca **[23]**.

Para conectarse al motor de base de datos

- En Management Studio, en el menú Archivo, hacer clic en Conectar Explorador de objetos **[23]**.
- Se abre el cuadro de diálogo Conectar al servidor. En el cuadro tipo de servidor se muestra el último tipo de componente utilizado **[23]**.
- Seleccionar Motor de base de datos **[23]**.

- En el cuadro nombre del servidor, escribir el nombre de la instancia de Motor de base de datos. Para la instancia predeterminada de SQL Server, el nombre de servidor es el nombre del equipo. Para una instancia con nombre de SQL Server, el nombre del servidor es <computer_name>\<instance_name>, como ACCTG_SRVR\SQLEXPRESS [23].
- Hacer clic en Conectar.

2.3.9 Autorizar conexiones adicionales

Una de las primeras tareas que deben realizarse es autorizar a otros usuarios a conectarse. Para ello, crear un inicio de sesión y conceder autorización para obtener acceso a una base de datos como usuario. Los inicios de sesión pueden ser inicios de sesión con autenticación de Windows, que utilizan las credenciales de Windows, o bien inicios de sesión con autenticación de SQL Server, que almacenan la información de autenticación en SQL Server y son independientes de las credenciales de Windows. Utilizar la autenticación de Windows siempre que sea posible [23].

2.3.10 Crear un inicio de sesión con autenticación de Windows

- En la tarea anterior, se conectó a Motor de base de datos utilizando Management Studio. En el Explorador de objetos, expandir la instancia de servidor, expandir Seguridad, hacer clic con el botón secundario en Inicios de sesión y, a continuación, hacer clic en Nuevo inicio de sesión [23].
- Aparecerá el cuadro de diálogo Inicio de sesión – Nuevo [23].
- En la página General, en el cuadro Nombre de inicio de sesión, escribir un inicio de sesión de Windows con el formato <domain>\<login> [23].
- En el cuadro Base de datos predeterminada, seleccionar la base de datos de elegida, si está disponible. Si no lo está, seleccionar master [23].
- En la página Roles del servidor, si el nuevo inicio de sesión va a ser administrador, hacer clic en sysadmin; de lo contrario, dejarlo en blanco [23].

- En la página Asignación de usuarios, seleccionar Asignar para la base de datos elegida, si está disponible. Si no lo está, seleccionar master. Observar que el cuadro Usuario se ha rellenado con el inicio de sesión. Al cerrar el cuadro de diálogo, se creará el usuario en la base de datos [23].
- En el cuadro Esquema predeterminado, escribir dbo para asignar el inicio de sesión al esquema del propietario de la base de datos [23].
- Aceptar los valores predeterminados de los cuadros Elementos protegibles y Estado y hacer clic en Aceptar para crear el inicio de sesión [23].

2.3.11 Conectarse desde otro equipo

Para mejorar la seguridad, no se puede obtener acceso a Motor de base de datos de las ediciones de SQL Server Developer, Express y Evaluation desde otro equipo cuando se instala inicialmente. En esta sección se muestra cómo habilitar los protocolos, configurar los puertos y configurar el Firewall de Windows para conectarse desde otros equipos [24].

2.3.12 Habilitar protocolos

Para mejorar la seguridad, SQL Server Express Developer y Evaluation se instalan con conectividad de red limitada. Las conexiones a Motor de base de datos se pueden realizar desde herramientas que se ejecuten en el mismo equipo, no desde otros equipos. Si se tiene previsto realizar las tareas de desarrollo en el mismo equipo que Motor de base de datos, no es necesario habilitar otros protocolos [24].

Si se tiene previsto conectarse al Motor de base de datos desde otro equipo, se debe habilitar un protocolo, como TCP/IP [24].

2.3.13 Habilitar conexiones TCP/IP desde otro equipo

- En el menú Inicio, elegir Todos los programas, Microsoft SQL Server 2012, Herramientas de configuración y, por último, Administrador de configuración de SQL Server [24].

- En Administrador de configuración de SQL Server, expandir Configuración de red de SQL Server y, a continuación, hacer clic en Protocolos de <InstanceName> [24].
- La instancia predeterminada (una instancia sin nombre) aparece como MSSQLSERVER. Si se ha instalado una instancia con nombre, el nombre proporcionado aparece en la lista. SQL Server 2012 Express se instala como SQLEXPRESS, a menos que se haya cambiado el nombre durante la instalación [24].
- En la lista de protocolos, hacer clic con el botón secundario en el protocolo que se desee habilitar (TCP/IP) y, a continuación, hacer clic en Habilitar [24].

Nota:

Se debe reiniciar el servicio SQL Server después de realizar los cambios en los protocolos de red.

2.3.14 Configurar un puerto fijo

Para mejorar la seguridad, Windows Server 2008, Windows Vista y Windows 7 activan el Firewall de Windows. Si se desea conectar a esta instancia desde otro equipo, es necesario abrir un puerto de comunicaciones en el firewall. La instancia predeterminada de Motor de base de datos escucha en el puerto 1433; por tanto, no se tiene que configurar un puerto fijo. No obstante, las instancias con nombre incluidas las de SQL Server Express escuchan en puertos dinámicos. Para poder abrir un puerto en el firewall, debe configurar primero Motor de base de datos para que escuche en un puerto específico conocido como puerto fijo o estático; de lo contrario, es posible que Motor de base de datos escuche en un puerto distinto cada vez que se inicie. Para obtener más información acerca de los firewall, de la configuración predeterminada de Firewall de Windows y una descripción de los puertos TCP que afectan al motor de base de datos, Analysis Services, Reporting Services e Integration Services, se debe Configurar Firewall de Windows para permitir el acceso a SQL Server [24].

Nota:

Las asignaciones del número de puerto son administradas por la Agencia de asignación de números Internet y se muestran en <http://www.iana.org>. Los números de puerto deben tener asignados números de 49152 a 65535 [24].

2.3.15 Configurar SQL Server para escuchar en un puerto específico

- En el Administrador de configuración de SQL Server, expandir Configuración de red de SQL Server y, a continuación, hacer clic en la instancia de servidor que se desee configurar [24].
- En el panel derecho, hacer doble clic en TCP/IP [24].
- En el cuadro de diálogo Propiedades de TCP/IP, hacer clic en la pestaña Direcciones IP [24].
- En el cuadro Puerto TCP de la sección IPAll, escribir un número de puerto disponible. Como ejemplo se podría utilizar 49172 [24].
- Hacer clic en Aceptar para cerrar el cuadro de diálogo y en Aceptar cuando aparezca una advertencia que indique que debe reiniciarse el servicio [24].
- En el panel izquierdo, hacer clic en Servicios de SQL Server [24].
- En el panel derecho, hacer clic con el botón secundario en la instancia de SQL Server y, a continuación, hacer clic en Reiniciar. Cuando se reinicia Motor de base de datos, escucha en el puerto 49172.
- Abrir puertos del firewall.
- Los sistemas de firewall ayudan a evitar el acceso no autorizado a los recursos de los equipos. Para conectarse a SQL Server desde otro equipo cuando el firewall está activo, se debe abrir un puerto en el firewall [24].

Nota:

El hecho de abrir puertos en el firewall puede dejar el servidor expuesto a ataques malintencionados. Asegúrese de que conoce los sistemas de firewall antes de abrir puertos. Para obtener más información, ver Consideraciones de seguridad para una instalación de SQL Server [24].

Después de configurar Motor de base de datos para utilizar un puerto fijo, seguir estas instrucciones para abrir ese puerto en el Firewall de Windows. (No es necesario configurar un puerto fijo para la instancia predeterminada, porque ya está fijada en el puerto TCP 1433).

Para abrir un puerto en el Firewall de Windows para el acceso TCP (Windows 7)

- En el menú Inicio, hacer clic en Ejecutar, escriba WF.msc y, a continuación, hacer clic en Aceptar [24].
- En la opción Firewall de Windows con seguridad avanzada del panel izquierdo, hacer clic con el botón secundario en Reglas de entrada y, a continuación, hacer clic en Nueva regla en el panel de acciones [24].
- En el cuadro de diálogo Tipo de regla, seleccionar Puerto y, a continuación, hacer clic en Siguiente [24].
- En el cuadro de diálogo Protocolo y puertos, seleccionar TCP. Seleccionar Puertos locales específicos y, a continuación, escribir el número de puerto de la instancia de Motor de base de datos. Usar 1433 para la instancia predeterminada. Escribir 49172 si está configurando una instancia con nombre y ha se configurado un puerto fijo. Hacer clic en Siguiente [24].
- En el cuadro de diálogo Acción, seleccionar Permitir la conexión y, a continuación, hacer clic en Siguiente [24].
- En el cuadro de diálogo Perfil, seleccionar los perfiles que describan el entorno de conexión del equipo cuando se desee conectar al Motor de base de datos y, a continuación, hacer clic en Siguiente [24].
- En el cuadro de diálogo Nombre, escribir un nombre y una descripción para esta regla. Después, haga clic en Finalizar [24].

2.3.16 Conectarse al motor de base de datos desde otro equipo

Cuando el servicio Explorador de SQL Server se ejecute en el equipo servidor y el firewall se haya abierto el puerto UDP 1434, la conexión se podrá realizar utilizando el nombre del equipo y el nombre de la instancia [24].

Para conectarse al motor de base de datos desde otro equipo

- En un segundo equipo que incluya las herramientas de cliente de SQL Server, iniciar una sesión con una cuenta autorizada para conectarse a SQL Server y abra Management Studio [24].
- En el cuadro de diálogo Conectar al servidor, confirmar Motor de base de datos en el cuadro Tipo de servidor [24].
- En el cuadro Nombre del servidor, escribir tcp: para especificar el protocolo, seguido del nombre del equipo, una coma y el número de puerto. Para conectarse a la instancia predeterminada, el puerto 1433 está implícito y se puede omitir, por lo que se deberá escribir tcp:<computer_name>. En el ejemplo de una instancia con nombre, escriba tcp:<computer_name>,49172 [24].

Nota:

Si se omite tcp: en el cuadro Nombre del servidor, el cliente probará todos los protocolos habilitados en el orden especificado en la configuración del cliente.

Finalmente en el cuadro Autenticación, confirmar Autenticación de Windows y, a continuación, hacer clic en Conectar [24].

2.3.17 Administración de la Base de Datos

Entre Herramientas de administración de SQL Server se incluyen las siguientes:

SQL Server Management Studio

SQL Server Management Studio (SSMS) es un entorno integrado para acceder, configurar, gestionar, administrar, y el desarrollo de todos los componentes de SQL Server. SSMS combina un amplio grupo de herramientas gráficas con un número de ricos editores de script para proporcionar acceso a SQL Server a los desarrolladores y administradores de todos los niveles [29].

SSMS combina las características de Enterprise Manager, Analizador de consultas y Analysis Manager, incluido en versiones anteriores de SQL Server, en un único entorno. Además, SSMS trabaja con todos los componentes de SQL Server, como Reporting Services e Integration Services. Los desarrolladores obtienen una experiencia familiar, y los administradores de bases de datos reciben una sola utilidad integral que combina herramientas gráficas fáciles de usar, con capacidades de scripting ricos [29].

Administrador de configuración de SQL Server

El Administrador de configuración de SQL Server es una herramienta para administrar los servicios asociados a SQL Server, para configurar los protocolos de red utilizados por SQL Server y para administrar la configuración de conectividad de red de los equipos cliente de SQL Server. El Administrador de configuración de SQL Server es un complemento de Microsoft Management Console que está disponible desde el menú Inicio o que se puede agregar a cualquier otra pantalla de Microsoft Management Console. Microsoft Management Console (mmc.exe) utiliza el archivo SQLServerManager10.msc de la carpeta System32 de Windows para abrir el Administrador de configuración de SQL Server [30].

El Administrador de configuración de SQL Server y SQL Server Management Studio utiliza el Instrumental de administración de Windows (WMI) para ver y cambiar algunas configuraciones del servidor. WMI proporciona una forma unificada de crear una interfaz con las llamadas a la API que administran las operaciones del Registro solicitadas por las herramientas de SQL Server y de proporcionar un mejor control y manipulación de los servicios SQL seleccionados del complemento del Administrador de configuración de SQL Server [30].

SMO

Gestión de objetos de SQL Server (SMO) es una colección de objetos que están diseñados para la programación de todos los aspectos de la gestión de Microsoft SQL Server. Servidor de administración de réplica SQL Objetos (RMO) es una colección de objetos que encapsula la administración de replicación de SQL Server [32].

SQLCMD

La utilidad sqlcmd permite introducir instrucciones de Transact-SQL, procedimientos del sistema y archivos de comandos en el símbolo del sistema, en el Editor de consultas en modo SQLCMD, en un archivo de comandos de Windows o en un sistema operativo (cmd.exe). Esta herramienta utiliza ODBC para ejecutar lotes de Transact-SQL [31].

2.3.18 Procesamiento de instrucciones SQL.

Los pasos básicos que SQL Server utiliza para procesar una única instrucción SELECT incluyen lo siguiente:

- El analizador examina la instrucción SELECT y la divide en unidades lógicas como palabras clave, expresiones, operadores e identificadores [33].
- Se genera un árbol de la consulta, a veces denominado árbol de secuencia, que describe los pasos lógicos que se requieren para transformar los datos de origen en el formato que necesita el conjunto de resultados [33].
- El optimizador de consultas analiza diferentes formas de acceso a las tablas de origen. A continuación, selecciona la serie de pasos que devuelve los resultados de la forma más rápida utilizando el menor número posible de recursos. El árbol de la consulta se actualiza para registrar esta serie exacta de pasos. La versión final y optimizada del árbol de la consulta se denomina plan de ejecución [33].
- El motor relacional comienza a ejecutar el plan de ejecución. A medida que se procesan los pasos que necesitan datos de las tablas base, el motor relacional solicita al motor de almacenamiento que pase los datos de los conjuntos de filas solicitados desde el motor relacional [33].
- El motor relacional procesa los datos que devuelve el motor de almacenamiento en el formato definido para el conjunto de resultados y devuelve el conjunto de resultados al cliente [33].

Los pasos básicos descritos para procesar una instrucción SELECT se aplican a otras instrucciones SQL como INSERT, UPDATE y DELETE. Las instrucciones UPDATE y DELETE deben identificar el conjunto de filas que se van a modificar o eliminar. El proceso de identificación de estas filas es el mismo que se utiliza para identificar las filas de origen que contribuyen al conjunto de resultados de una instrucción SELECT. Las instrucciones UPDATE e INSERT pueden contener instrucciones SELECT incrustadas que proporcionan los valores de los datos que se van a actualizar o insertar [33].

Incluso las instrucciones del lenguaje de definición de datos (DDL), como CREATE PROCEDURE o ALTER TABLE, se resuelven al final en un conjunto de operaciones relacionales en las tablas de catálogo del sistema y, a veces (como ALTER TABLE ADD COLUMN) en las tablas de datos [33].

2.3.19 Creación de base de datos en SQL Server

Usando SQL Server Management Studio [26]

- En el Explorador de objetos, conectarse a una instancia del Motor de base de datos de SQL Server y expandirla.
- Hacer clic con el botón secundario en Bases de datos y, a continuación, en Nueva base de datos.
- En Nueva base de datos, especificar un nombre de base de datos.
- Si se desea crear la base de datos aceptando todos los valores predeterminados, hacer clic en Aceptar; de lo contrario, continuar con siguientes los pasos opcionales.
- Para cambiar el nombre de nombre del propietario, hacer clic en (...) para seleccionar otro.
- Para cambiar los valores predeterminados de los archivos de datos y de registro de transacciones principales, en la cuadrícula Archivos de la base de datos, hacer clic en la celda correspondiente y especifique el nuevo valor. Para obtener más información, ver Agregar archivos de datos o de registro a una base de datos.

- Para cambiar la intercalación de la base de datos, seleccionar la página Opciones y una intercalación de la lista.
- Para cambiar el modelo de recuperación, seleccionar la página Opciones y un modelo de recuperación de la lista.
- Para cambiar opciones de base de datos, seleccionar la página Opciones y modificar las opciones de la base de datos. Para obtener una descripción de cada una de las opciones, ver Opciones de ALTER DATABASE SET (Transact-SQL).
- Para agregar un nuevo grupo de archivos, hacer clic en la página Grupos de archivos. Hacer clic en Agregar y especifique los valores para el grupo de archivos.
- Para agregar una propiedad extendida a la base de datos, seleccionar la página Propiedades extendidas.
- En la columna Nombre, escribir un nombre para la propiedad extendida.
- En la columna Valor, escribir el texto de la propiedad extendida. Por ejemplo, especificar una o varias instrucciones que describan la base de datos.
- Para crear la base de datos, haga clic en Aceptar.

2.4 EL RENDIMIENTO EN LOS SGBD

A continuación se hace referencia a la definición de rendimiento.

En informática, medida o cuantificación de la velocidad/resultado con que se realiza una tarea o proceso. En una computadora, su rendimiento no depende sólo del microprocesador como suele pensarse, sino de la suma de sus componentes como la memoria, el bus, los diversos dispositivos, etc. y su software [1] [6].

El objetivo de evaluar el rendimiento de un SGBD se debe enfocar de manera constante al funcionamiento del SGBD, para optimar el uso de los recursos de hardware , como uso de CPU, uso de Memoria RAM y uso de Red, para ver su desempeño [6].

Debido que si se detecta que las transacciones no son respondidas con la rapidez que se necesita o se pierden datos, el administrador de base de datos, debe aplicar las medidas

correctivas para solucionar las fallas, analizando su comportamiento a través de herramientas gráficas que existen en el mercado [6].

Luego de revisar las definiciones de algunos autores que hablan sobre el rendimiento de las bases de datos, los autores de esta investigación concuerdan en el hecho de que el rendimiento está estrictamente relacionado con la velocidad de respuesta y el uso óptimo de los recursos del equipo en el que está trabajando el SGBD ante la carga de trabajo al que está expuesto.

CAPÍTULO III

3. ESTUDIO DE HERRAMIENTAS Y DEFINICIÓN DE MÉTRICAS

3.1 Investigación de herramientas para medir el rendimiento de los SGBD

Para la presente investigación se necesita de herramientas que faciliten la medición del rendimiento de los SGBD, por lo cual se va a considerar dos tipos de herramientas. La primera permitirá realizar un conjunto de transacciones hacia la base de datos y la segunda monitoreará a cada uno de los servidores de base de datos, permitiendo de esta manera obtener resultados de forma cuantitativa de todos los recursos de hardware que se verán involucrados durante la ejecución de las transacciones hacia los SGBD.

Existe una gran variedad de herramientas de código abierto y bajo licencia, de esta manera se va a plantear a continuación estas herramientas con sus características sobresalientes, de tal manera de hacer la mejor elección y posteriormente hacer uso de ellas para poder determinar cuál SGBD brinda un mejor rendimiento, el cual es el objetivo de este proyecto.

3.1.1 Herramientas de pruebas de carga

Existen varias herramientas en el mercado que permiten ejecutar pruebas de carga, para lo cual se revisó las características de cada una de ellas y se han elegido las que según el criterio de los autores se enfocan al proyecto de estudio, entre ellas se destacan: Sql Query Stress, Hammerdb, Advanced Query Tool, Database Stress Testing Tool.

❖ SQL Query Stress

Es una herramienta gratuita para programadores de SQL Server. Está diseñado para ayudar con las pruebas de estrés de rendimiento de las consultas T-SQL y rutinas. Proporciona pruebas de rendimiento simple, de manera ligera, diseñado para cargar las consultas individuales de las pruebas. Incluye soporte para la asignación al azar de los parámetros de entrada con el fin de probar la repetitividad de caché, e incluye capacidades básicas para la presentación de informes sobre los recursos del servidor que se consumen [19].

La **figura III.6** ilustra la interfaz de la herramienta.

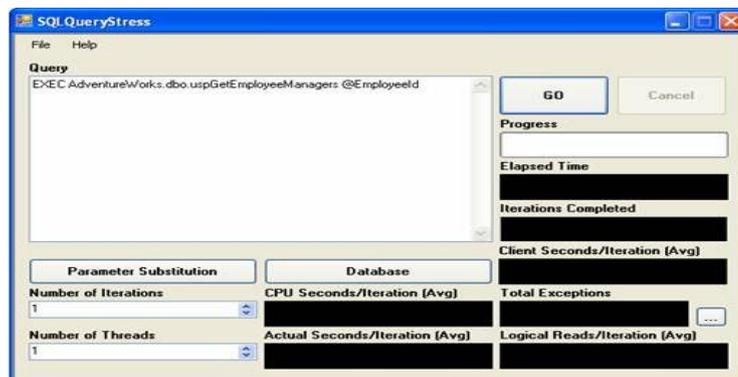


Figura III.6: Herramienta SQL Query Stress

Fuente: <http://www.datamanipulation.net/sqlquerystress/>

De manera general la aplicación funciona de la siguiente manera:

- **El área de consultas:** es donde se puede ingresar la consulta que se desea cargar en la prueba. Esto puede ser una consulta T-SQL o una llamada a un procedimiento almacenado. También puede incluir los nombres de variables, que pueden ser utilizados para la sustitución de parámetros dinámicos [19].
- **El cuadro Número de iteraciones:** es donde se puede definir el número de veces que la consulta debe ser ejecutada por cada usuario virtual como se define en el cuadro *Número de Hilos*. Se puede especificar un máximo de 200 hilos, que se desarrollará la consulta simultáneamente con el fin de simular la carga [19].
- **El botón GO:** se iniciará la prueba de carga. Durante la prueba de la *barra de progreso* indica el número de pruebas realizadas en comparación con el número de pruebas que se ejecutarán [19].
- **Las iteraciones completas:** muestra cuántas pruebas han sido realmente ejecutadas [19].
- **Excepciones:** indica la cantidad de excepciones que han ocurrido durante la ejecución [19].

Hay tres tipos de estadísticas sobre el tiempo recogidas por la herramienta.

- **El cuadro Segundos cliente / Iteración:** muestra el tiempo de ejecución promedio de todas las iteraciones, según registre en el cliente [19].
- **Los segundos de CPU / Iteración y Actual segundos / iteración.** El primero corresponde al promedio reportado del tiempo de CPU, y el segundo es el promedio reportado del tiempo total de la consulta [19].

- **Número de lecturas lógicas:** que es una fusión de caché de búfer y lecturas de disco [19].

Como se puede observar esta herramienta provee un tipo de pruebas muy básicas y solo orientado a un tipo de base de datos como lo es SQL Server.

❖ **Hammerdb**

Es una herramienta de carga de código abierto orientada a medir el benchmarking para base de datos como: Oracle, SQL Server, TimesTen, PostgreSQL, Greenplum, Postgres Plus Advanced Server, MySQL y Redis [14].

Algunas características que proporciona la herramienta son [14]:

- Permite comparar base de datos y sistemas
- Automatización de las pruebas
- Administra usuarios virtuales - multi-hilo
- Mide las pruebas de carga y la carga de trabajo en repetición
- Construye pruebas transaccionales y consultas basadas en parámetros estándares de la industria

Esta herramienta permite que se puedan realizar varios tipos de pruebas como [14.]:

- Pruebas OLTP - basada en la especificación TPC-C
- Pruebas de DSS - basada en la especificación TPC-H
- Pruebas Web
- Autohammer

La **figura III.7** ilustra la interfaz gráfica de la herramienta

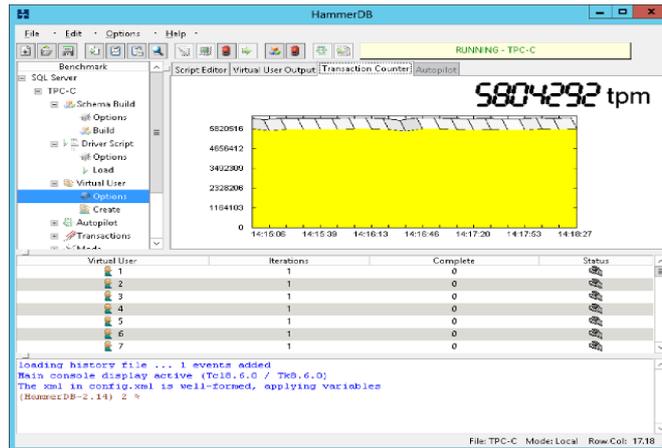


Figura III.7: Herramienta Hammerdb

Fuente: <http://hammerora.sourceforge.net/>

Existen algunos estudios que han confiado en la herramienta Hammerdb para realizar pruebas de carga, en **Tabla III.VIII** se nombra a algunos de ellos.

Tabla III.VIII: Estudios y White papers que han usado Hammerdb

Fuente: <http://www.hammerdb.com/community.html>

Estudios y White papers que utilizaron Hammerdb

2014

- intel Oracle Openworld Keynote 2014 Silicon Photonics with HammerDB (@35-40 mins)
- An Approach for Hybrid-Memory Scaling Columnar In-Memory Databases (SAP HANA with HammerDB)
- High Performance Oracle Workloads with the Dell Acceleration Appliances for Databases
- Having Your Cake and Eating it: Affordable Performance with HP and Fusion-io Atomic Series
- Accelerating Virtualized Oracle 12c performance with Vsphere 5.5 Advanced Features Flash Read Cache and Vmotion
- Mission Critical Database performance: Intel Xeon Processor E7 v2 family vs.IBM Power7+
- Microsoft SQL Server 2012 with AlwaysOn Using Cisco UCS C-Series Servers and Fusion ioMemory

- Fusion ioMemory and Hyper-V Workload using Windows Server 2012 Hyper-V and SQL Server 2012 Performance Whitepaper
- Dell Poweredge R920 running Oracle Database:benefits of upgrading with NVM Express Flash PCIe SSDs
- Dual parallelism: Intel NVMe SSD and Oracle TimesTen
- Microsoft SQL Server Acceleration Flash Buyer's Guide
- Oracle RAC 12c Reference Architecture with ION Accelerator
- Intel® Xeon® Processor E7-4800 v2 Product Family (Four-socket business processing performance benchmarks)
- ServerSmart - Chat 1 Recap: Datacenter Modernization
- Product Review: HP ProLiant DL580 Gen8
- Dell Fluid Cache for SAN - Principled Technologies
- PSAM; NEC PCIe SSD Appliance for Microsoft® SQL Server
- Proven Performance as a High Performance Microsoft SQL Server Database Solution Combining an NEC Express 5800 Scalable Enterprise Server and HGST FlashMAX II
- Database Acceleration with HP 3PAR StoreServ 7450 All-flash Storage
- SQL Server on K2
- HGST Shows Off Zippy PCM Prototype, Updates Flash Cards

2013

- Accelerate Applications Using EqualLogic Arrays with directCache
- TechEd 2013 Demo 1: 69 Billion Transactions Fusion-io
- TechEd 2013 Demo 2: Simplify Cluster Scale Out Fusion-io
- ION performance brief hp dl980-8b
- Lenovo Summit 2013 Demo: Lenovo High Performance SQL Server Solution
- Transforming Oracle Applications with Flash-Optimized Snapshots
- Marvell DragonFly With SanDisk® X110 SSD 256GB TPC-C Benchmark Test Results
- HDS UCP for Oracle – How does it stack up?
- sTec to Demonstrate the Performance Benefits of its Enterprise-Class Solid-State Storage Solutions at Percona Live MySQL Conference & Expo
- sTec Host Cache Solution - NetApp Solution Connection
- PostgreSQL on IBM Power
- Evaluation of In-memory database TimesTen
- Nutanix Microsoft SQL Server Best Practices
- HP Universal Solutions for Oracle

2012

- Marvell DragonFly NV DRAM Storage Accelerator
- Nobody Got Fired for Putting Oracle Redo on Pure Storage
- Open Source Solutions Initiative

❖ **Advanced Query Tool (AQT)**

Es una herramienta rápida y eficaz de consultas para múltiples bases de datos, proporcionando una interfaz ODBC de bajo nivel que se ejecuta de forma fiable contra la base de datos pequeñas hasta los más grandes sistemas de una empresa [5].

Las siguientes características de la herramienta de consulta avanzada son [5]:

- Permite ver los objetos en su base de datos
- Ejecutar sentencias SQL
- Visualizar los datos
- Actualizar sus datos
- crear consultas con la gráfica de generador de consultas
- Exportar sus datos
- comparar los datos - incluso a través de los tipos de bases de datos
- Administrar sus bases de datos - un *front-end* para los diferentes tipos de bases de datos
- Carga datos en las tablas - incluso a través de los tipos de bases de datos
- Comparar objetos - comparar la definición de las tablas y otros objetos
- Monitorea el sistema

La **figura III.8** ilustra la interfaz gráfica de la herramienta.

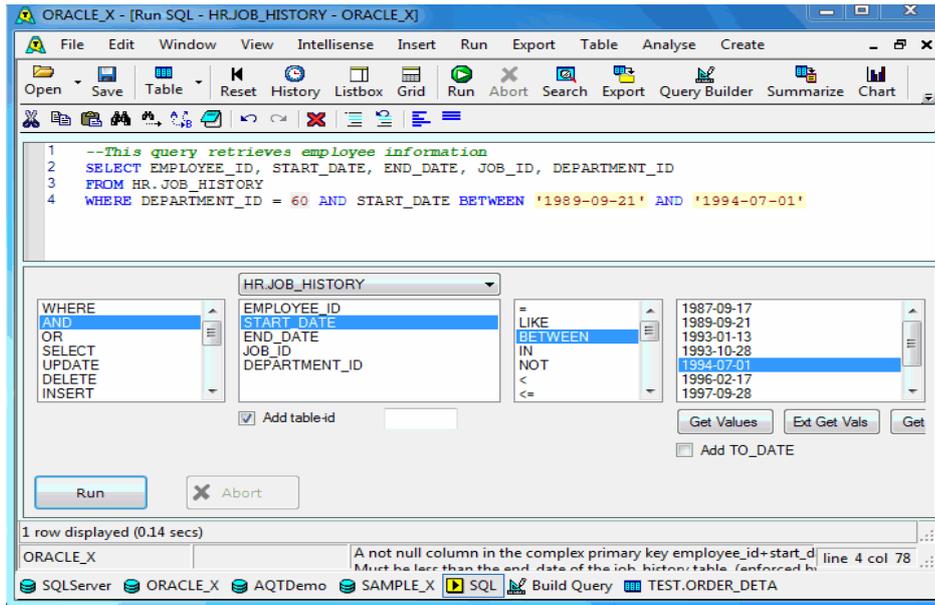


Figura III.8: Herramienta Advanced Query Tool (AQT)

Fuente: <http://www.querytool.com/>

❖ Database Stress Testing Tool

Es un software para pruebas de estrés y pruebas de carga para servidores de sistemas de información y aplicaciones de bases de datos [9].

La herramienta permite a los usuarios crear y configurar un conjunto continuo de peticiones al servidor del OLAP (ejecución de la consulta) y OLTP (añadir / cargar, modificar y eliminar datos en la base de datos). El usuario puede especificar el número de usuarios virtuales para ser emulados, prioridad y el tipo de solicitudes de cada tarea (tipo de usuario virtual) [9].

La herramienta de estrés es útil para cualquier desarrollador de sistemas de información o de control de calidad empleado del departamento que quiere asegurarse de que el producto en fase de desarrollo puede trabajar con un gran número de conexiones simultáneas o concurrentes consultas y transacciones. Administradores de bases de

datos pueden utilizar esta utilidad para evaluar el actual desempeño de los servidores y para hacer planes sobre el aumento de potencia de procesamiento [9].

Principales características de la herramienta [9]:

- Reportes de las pruebas de ejecución en formato HTML o Microsoft Excel.
- Base de datos independiente, la herramienta es compatible con todas las interfaces de estrés de bases de datos unificadas: ODBC, IDAPI, OLE DB y Oracle Call Interface.
- Soporte de sentencia de SQL de manera dinámica lo que permite realizar pruebas de carga más flexibles.
- Soporte de línea de comandos (modo de consola) ayuda a los usuarios a automatizar el proceso de pruebas de carga
- Permite importar herramienta de SQL Server Profiler y así hacer la definición de tareas que sea más realista.
- Provee un asistente para la comprobación de varios informes, luego de las pruebas previamente realizadas
- DTM DB Stress es una aplicación de 32 bits compatible con Windows XP/Vista/7, Windows 8 (de escritorio), Windows Server 2003/2008/2012. La edición de 64 bits está disponible a petición para los usuarios registrados.

La **figura III.9** ilustra la interfaz gráfica de la herramienta

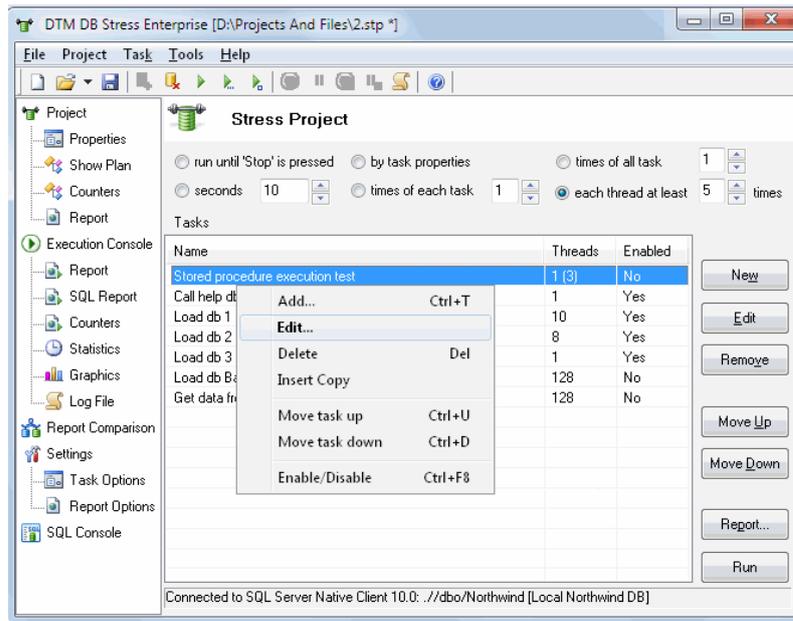


Figura III.9: Herramienta Database Stress Testing Tool

Fuente: <http://www.sqledit.com/stress/>

3.1.2 Herramientas de monitoreo

Existen varias herramientas en el mercado que permiten monitorear el uso de recursos de los servidores de base de datos, de esta manera se procedió a investigar algunas herramientas con sus respectivas características y posteriormente según el criterio de los autores se ha elegido las que ofrezcan mejores prestaciones como: portabilidad y usabilidad.

Las herramientas a estudiar son las siguientes: Manage Engine, Windows Task Manager, Performance Advisor for SQL Server.

- **Manage Engine**

Es un sistema de monitorización que ofrece información sobre disponibilidad y rendimiento de aplicaciones de negocio desde 3 enfoques diferentes [16]:

- **Monitorización de componentes de aplicaciones:** con el fin de identificar problemas específicos de servidores de aplicaciones web, bases de datos,

servidores críticos, sitios web, servicios, etc., así como aplicaciones a medida. Los componentes pueden agruparse en grupos lógicos e identificar la causa raíz de los problemas en sus aplicativos [16].

- **Monitorización de aplicaciones a nivel transaccional**, identificando el tiempo de respuesta y el rendimiento de las mismas [16].
- **Monitorización de experiencia de usuario final EUM**, que permite sondear servicios o aplicaciones web desde localizaciones externas a la red empresarial [16].

Ofrece algunas funcionalidades como las siguientes:

- **Descubrimiento de aplicaciones:** Descubre automáticamente los servidores de aplicaciones, bases de datos, servicios, sistemas, transacciones, recursos virtuales, etc, [16].
- **Vista global de los recursos TI:** Gestiona un amplio rango de aplicaciones de negocio y servicios de red agrupándolos en unidades lógicas de negocio [16].
- **Gestión de fallos:** Identifica fallos de las aplicaciones en base a disponibilidad, umbrales o anomalías y automatiza la respuesta a los problemas ejecutando programas/scripts, servicios Windows, máquinas virtuales [16].

A la vez ofrece un amplio conjunto de monitores de aplicaciones que incluyen alrededor de 100 parámetros críticos como tiempo de respuesta, disponibilidad, utilización de CPU, etc [16].

- Monitorización de plataformas y tecnologías
- Monitorización de bases de datos
- Monitorización de servidores y sistemas críticos
- Monitorización específica de sistemas ERP
- Monitorización de virtualización

- Monitorización de servidores y servicios web
- Monitorización de sitios web
- Monitorización de Middleware/Portal
- Monitorización en la nube
- Monitorización de usuarios finales (EUM)
- Monitorización de transacción web

En base a la lista anterior se va hacer uso de las herramientas de monitorización de base de datos, cabe recalcar que existen herramientas de monitorización propias para cada base de datos, por lo que se utilizaría: *PostgreSQL Monitoring* y *SQL Server Monitoring*

Algunos de los componentes que son monitoreados en las bases de datos son:

- Uso de la memoria
- Estadísticas Buffer Manager
- Estadísticas de conexión
- Detalles de caché
- Detalles Lock
- Estadísticas de SQL
- Detalles Latch
- Método de acceso Detalles
- Detalles de bases de datos
- Trabajos programados

La **figura III.10** ilustra a manera de ejemplo la forma como la herramienta muestra los porcentajes de uso de los distintos recursos.

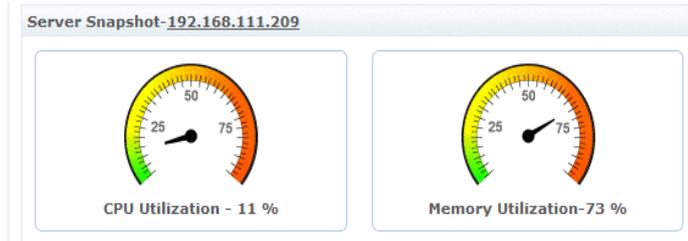


Figura III.10: Herramienta Manage Engine

Fuente: <http://www.Manage Engine.com/>

Existen numerosas empresas que han confiado en la herramienta Manage Engine y destacan su fiabilidad para el monitoreo de sus instalaciones, a continuación en la **Tabla III.IX** se nombra a algunas de ellas.

Tabla III.IX: Empresas que utilizan Manage Engine

Fuente: http://www.manageengine.com/products/applications_manager/customers.html

Área Financiera	Área de la Salud	Servicios IT
<ul style="list-style-type: none"> • GE Money • JPMorganChase • MetLife • Dresdner-Cetelem Kreditbank • Deutsche Bank • First National Bank of Arizona • European Transaction Bank • Banco Central de Venezuela • Greenwich Associates • Genworth Financial • Universal Underwriters Group • NamClear 	<ul style="list-style-type: none"> • Pfizer • Boehringer Ingelheim • Lexicon-Genetics • Health Canada • City of Hope Medical Center • Wyeth • Achieve HealthCare • IndependentHealth • Wellinx • Sohard AG • Genova Diagnostics • Chemical Abstracts Services • Health Market Science 	<ul style="list-style-type: none"> • EDS • IBM • McKesson • Amazon • The Seemis Group • T&F Informa UK Limited • Türmob • FluidMedia • Webrizon • Osiris Trading (Pvt) Ltd • GalaxyPlus • Intelliquip, LLC • Infotrieve Inc • Model N • MetaFile • Onewebhosting.com

<ul style="list-style-type: none">• Aflains• Quaestor• NewBucks Operations• Metropolitan Life• Liberty International• Northern Rock Building Society	<ul style="list-style-type: none">• DPC Instrument Systems• Abbott• Sanofi Aventis• CH de Valenciennes• PAREXEL• Medtronic• Cystic Fybrois Trust• Landmark Medical Center	<ul style="list-style-type: none">• Forest Technology Group• ABEBooks• Jatis Solutions• Global Entertainment Inc• FuelQuest• Pearson Performance Solutions
---	--	---

❖ Windows Task Manager

El Administrador de tareas muestra los programas, procesos y servicios que se están ejecutando en un determinado equipo. Proporciona información limitada sobre el rendimiento del equipo y uso del CPU, memoria ram, disco duro, red, por la ejecución de aplicaciones y procesos.

La **figura III.11** ilustra la interfaz gráfica de la aplicación.

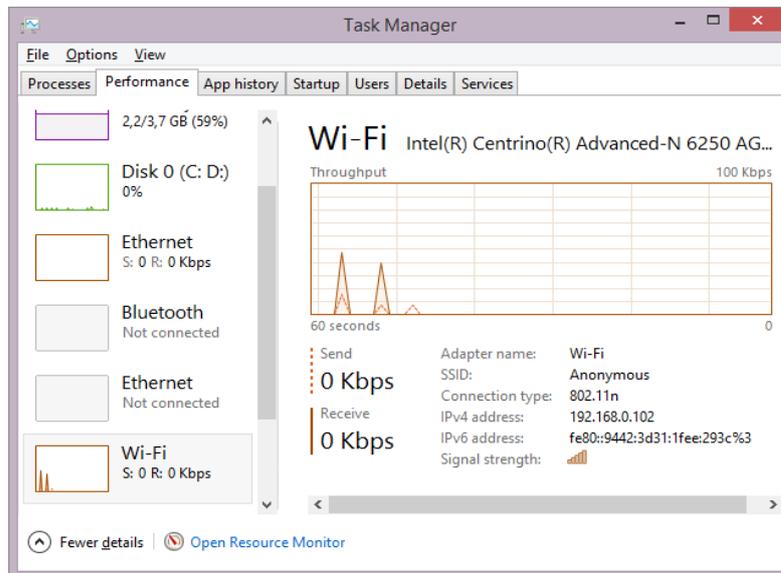


Figura III.11: Herramienta Task Manager

Fuente: http://en.wikipedia.org/wiki/Windows_Task_Manager

❖ **Performance Advisor for SQL Server**

Proporciona una visión sin precedentes, la conciencia y el control sobre la verdadera fuente de los problemas de rendimiento de SQL Server. Performance Advisor está repleto de características innovadoras que no se encuentran en ningún otro software de supervisión del rendimiento, diseñada con el único objetivo de simplificar el proceso de ajuste de rendimiento de SQL Server y la optimización.⁵

Performance Advisor utiliza el mismo cliente de gestión y monitorización de servicios como director de eventos para proporcionar la supervisión perfectamente integrada de SQL Server y la optimización, ajustando las consultas, alertas, y las características de planificación de tareas. Esto proporciona una variedad de beneficios, incluyendo un paquete unificado y único de instalación, configuración aerodinámica, y la capacidad de ver los eventos de rendimiento como bloques de SQL Server y los callejones sin salida en el calendario Event Manager.

La **figura III.12** ilustra las salidas del consumo de los recursos que proporciona la herramienta.



Figura III.12: Performance Advisor for SQL Server

Fuente: <http://www.sqlsentry.com/>

⁵ Información obtenida de: <http://www.sqlsentry.com/products/performance-advisor/sql-server-performance>

3.2 Selección de las herramientas

A continuación se presenta un estudio de cada una de las herramientas antes mencionadas, de donde se tomará en cuenta algunas variables, las cuales según el criterio de los autores de esta investigación son importantes al momento de elegir cada uno de ellas.

La **tabla III.X** representa las herramientas que realizan pruebas de carga sobre los servidores de base de datos, donde se marcará con una “x” las que cumplan las características planteadas.

Tabla III.X: Herramientas de pruebas de carga en BD

Fuente: Johnny Parra, Mario Andrade

CARACTERISTICAS	SQL QUERY STRESS	HAMMERDB	AQT	DTM DB STRESS
Bajo licencia libre	x	x		
Pruebas oltp basadas en tpc-c		x		x
Pruebas dss basadas en tpc-h		x		
Pruebas t-sql	x			
Manuales de referencia	x	x	x	x
Plataforma Windows	x	x	x	x
Plataforma Linux		x		
Arquitectura x64		x		x
Arquitectura x32	x	x	x	x
Portabilidad		x	x	x
total	5	9	4	5

En base a los resultados de la **tabla III.X** se puede determinar que la herramienta Hammerdb ofrece más características, de esta manera se podrá obtener resultados más óptimos y confiables durante la etapa de pruebas de carga.

La **tabla III.XI** contiene las herramientas que proporcionarán gráficos estadísticos sobre el consumo de recursos, luego de realizar la carga de trabajos efectuados por las herramientas previamente descritas en la **Tabla III.X**.

Tabla III.XI: Herramientas para monitorear el consumo de recursos

Fuente: Johnny Parra, Mario Andrade

CARACTERÍSTICAS	Manage Engine	Windows task manager	Performance Advisor for SQL Server
Bajo licencia libre		x	
Plataforma Windows	X		x
Plataforma Linux	X		
Portabilidad (soporte varios SGBD)	X	x	
TOTAL	3	2	1

En base a los resultados de la **tabla III.XI** se puede determinar que la herramienta Manage Engine ofrece más características lo cual permitirá obtener resultados más confiables durante la consulta de recursos consumidos por los servidores de base de datos.

3.3 Definición de las métricas

Los sistemas de gestión de bases de datos con los cuales se realizará el análisis comparativo son SQL Server 2012 Express Edition y PostgreSQL 9.2.4

En la ejecución de las pruebas se eligió varias métricas como son: Uso de CPU, Uso de RAM y Uso de Red, las cuales servirán para determinar cuál SGBD es mejor. A continuación se detallan las mismas.

3.3.1 Consumo de recursos

Los SGBD consumen recursos como CPU, RAM y en un ambiente distribuido se hace uso de la RED, las cuales constituyen en métricas importantes en el estudio del rendimiento.

Para la presente investigación se va hacer un monitoreo de las variables: uso de la memoria RAM, el uso de la CPU y uso de la red [6].

- **Uso de CPU**

El CPU es el responsable de ejecutar los comandos enviados por los programas de software para realizar una determinada función, de esta manera se resalta su importancia al momento de realizar las transacciones enviadas hacia el motor de base de datos, el mismo que deberá resolverlas en el menor tiempo posible.

Un uso excesivo continuo de la CPU puede indicar que las consultas de diferentes transacciones SQL deben optimizarse, o exista una sobrecarga de peticiones, como puede ser también que sea necesaria una actualización de la CPU.

- **Uso de Memoria RAM**

La memoria RAM constituye un espacio de almacenamiento temporal utilizado por el microprocesador y otros componentes, los SGBD la utilizan para guardar de manera temporal las consultas previamente ejecutadas aprovechando la velocidad de recuperación de los datos almacenados.

Si no se asignó o no hay disponible suficiente memoria para la BD, el rendimiento disminuirá. Los datos se deben leer en el disco, y no directamente en la caché de datos. Los sistemas operativos realizan una paginación excesiva intercambiando datos con el disco cuando son necesarias las páginas.

- **Uso de Red**

Al trabajar en un ambiente distribuido se debe tomar en cuenta la cantidad de tráfico que se genera por cada uno de los usuarios, con el fin de prevenir posibles cuellos de botella, sobrecarga en el uso de la red. Las transacciones realizadas por el cliente pueden tardar más tiempo debido a una congestión de la red.

3.3.2 Tiempo de respuesta

El tiempo requerido al ejecutar transacciones en una base de datos, es un elemento primordial al trabajar con volúmenes grandes de información, el SGBD más óptimo será aquel que ofrezca los mejores tiempos de respuesta al ejecutar los diferentes tipos de transacciones.

- Inserciones
- Consultas con un filtro

3.3.3 Número de órdenes por minuto

Es la capacidad de procesar cierto número de transacciones en un determinado intervalo de tiempo. A mayor número de órdenes (transacciones) realizadas por minuto se podrá determinar el mejor SGBD.

3.3.4 Tiempo del aseguramiento de la integridad de datos

La capacidad de comprobar la integridad de datos en un menor tiempo determinará el mejor SGBD.

La exigencia de integridad de los datos garantiza la consistencia de los datos de la base de datos. Por ejemplo, si se especifica para un empleado el valor de identificador de 567, la base de datos no debe permitir que ningún otro empleado tenga el mismo valor de identificador. Si tiene una columna `employee_rating` para la que se prevean valores entre 10 y 20, la base de datos no debe aceptar valores fuera de ese intervalo. Si en la tabla hay una columna `dept_id` en la que se almacena el número de departamento del empleado, la base de datos sólo debe permitir valores que correspondan a los números de departamento de la empresa **[25]**.

La integridad de datos pertenece a una de las siguientes categorías:

Integridad de entidad

La integridad de entidad define una fila como entidad única para una tabla determinada. La integridad de entidad exige la integridad de las columnas de los identificadores o la clave principal de una tabla, mediante índices y restricciones UNIQUE, o restricciones PRIMARY KEY [25].

Integridad de dominio

La integridad de dominio viene dada por la validez de las entradas para una columna determinada. Puede exigir la integridad de dominio para restringir el tipo mediante tipos de datos, el formato mediante reglas y restricciones CHECK, o el intervalo de valores posibles mediante restricciones FOREIGN KEY, restricciones CHECK, definiciones DEFAULT, definiciones NOT NULL y reglas [25].

Integridad referencial

La integridad referencial protege las relaciones definidas entre las tablas cuando se crean o se eliminan filas. La integridad referencial se basa en las relaciones entre claves externas y claves principales o entre claves externas y claves exclusivas, mediante restricciones FOREIGN KEY y CHECK. La integridad referencial garantiza que los valores de clave sean coherentes en las distintas tablas. Para conseguir esa coherencia, es preciso que no haya referencias a valores inexistentes y que, si cambia el valor de una clave, todas las referencias a ella se cambien en consecuencia en toda la base de datos [25].

Cuando se exige la integridad referencial, se impide a los usuarios:

- Agregar o cambiar filas en una tabla relacionada si no hay ninguna fila asociada en la tabla principal.
- Cambiar valores en una tabla principal que crea filas huérfanas en una tabla relacionada.
- Eliminar filas de una tabla principal cuando hay filas relacionadas coincidentes.

Por ejemplo, en las tablas Sales.SalesOrderDetail y Production.Product de la base de datos AdventureWorks2008R2, la integridad referencial se basa en la relación entre la clave externa (ProductID) de la tabla Sales.SalesOrderDetail y la clave principal (ProductID) de la tabla

Production.Product. Esta relación garantiza que un pedido de ventas no pueda nunca hacer referencia a un producto que no existe en la tabla Production.Product [25].

La **figura III.13** representa la integridad referencial mediante claves externas/principales

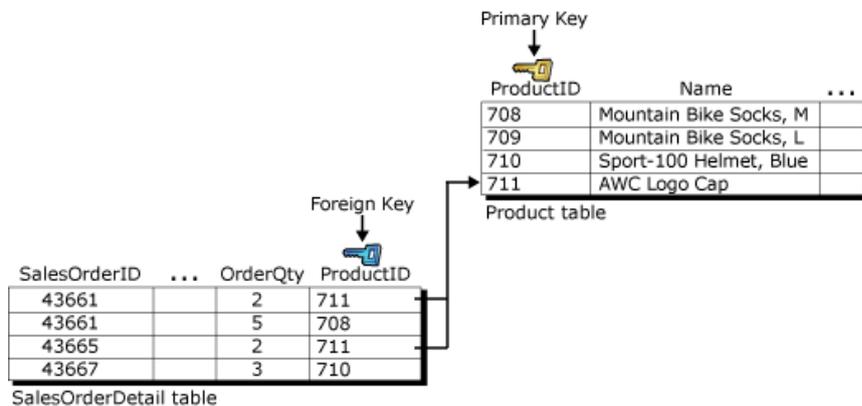


Figura III.13: Ejemplo integridad referencial

Fuente: [http://technet.microsoft.com/es-es/library/ms184276\(v=sql.105\)](http://technet.microsoft.com/es-es/library/ms184276(v=sql.105))

De igual manera se puede comprobar la integridad mediante el uso de un Trigger en la BD, el mismo que permita generar un valor auto calculado, luego de promediar los valores de algunos registros [25].

3.4 Determinación de los escenarios de prueba

Con el objetivo de que las pruebas se realicen en un entorno de total igualdad para ambos SGBS, la arquitectura de red que se requiere es de un servidor de base de datos al que lo vamos a llamar Sistema bajo pruebas, el cual estará instalado y configurado con los SGBD SQL Server y PostgreSQL, las pruebas se ejecutarán utilizando 2 máquinas en un entorno de red sencillo, conectando los dispositivos mediante un switch, tanto el SGBD SQL Server como PostgreSQL.

Para las pruebas de consumo de recursos y número de órdenes por minuto se requiere un equipo que genere las transacciones mediante el Software Hammedb, el mismo que simula un sistema de cómputo para colocar las órdenes, los pagos, la entrega de los pedidos y la capacidad de consultar los niveles de existencias de los almacenes. En consecuencia, la carga de trabajo se

define por una mezcla de 5 transacciones seleccionadas al azar de acuerdo con el equilibrio del valor de porcentaje se muestra de la siguiente manera:

- Nuevo orden: recibir un nuevo pedido de un cliente: 45%.
- Pago: actualizar el saldo de los clientes para registrar un pago: 43%.
- Entrega: entregar los pedidos de forma asíncrona: 4%.
- Orden-status: recuperar el estado de orden más reciente del cliente: 4%.
- Nivel de archivo: devolver el estado de las existencias en almacén: 4%.

Por otra parte la aplicación encargada del monitoreo será instalada en una misma máquina donde están instalados los SGBD, el generador de transacciones para los SGBD se instalará en otra máquina

Para las pruebas de tiempos de respuesta y Tiempo del aseguramiento de la integridad de datos se realizará una aplicación web sencilla, mediante la cual se enviará un número determinado de transacciones a los SGBD.

La **figura III.14** ilustra el esquema en el cual se realizarán las pruebas.

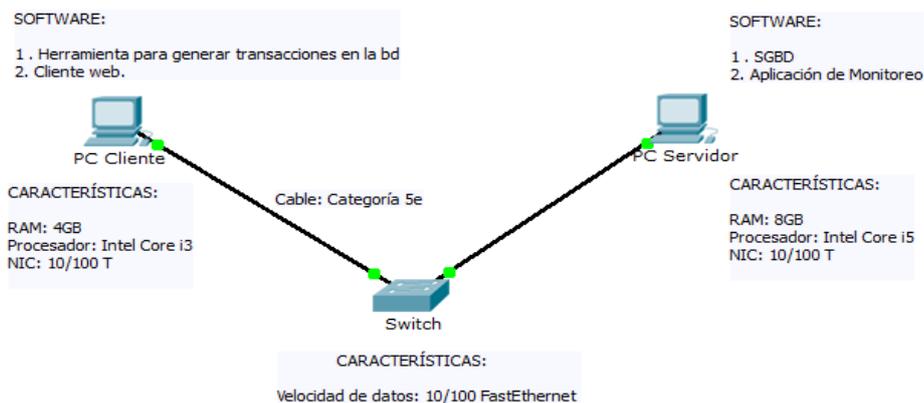


Figura III.14: Esquema utilizado para la realización de las pruebas

Fuente: Johnny Parra, Mario Andrade

3.4.1 Esquemas de Base de datos para las pruebas

Para las pruebas de consumo de recursos y número de órdenes por minuto se va a utilizar una base de datos con la misma estructura, cantidad de tablas, tipos de datos, llaves, en ambos SGBD, se ingresará el mismo volumen de información y longitud en cada uno de los datos. Previa la realización de cada una de las pruebas se utilizará la cantidad de registros indicados en la **Tabla III.XII**.

Tabla III.XII: Tablas y cantidad de registros usados en las pruebas de CR y NOPM

Fuente: Johnny Parra, Mario Andrade

POSTGRESQL Y SQLSERVER	
TABLAS	REGISTROS
CUSTOMER	1920000
DISTRICT	640
HISTORY	1920000
ITEM	100000
NEW_ORDER	576000
ORDER_LINE	19194711
ORDERS	1920000
STOCK	6400000
WAREHOUSE	64
Total	32031351

La **figura III.15** ilustra la estructura de la base de datos para las pruebas

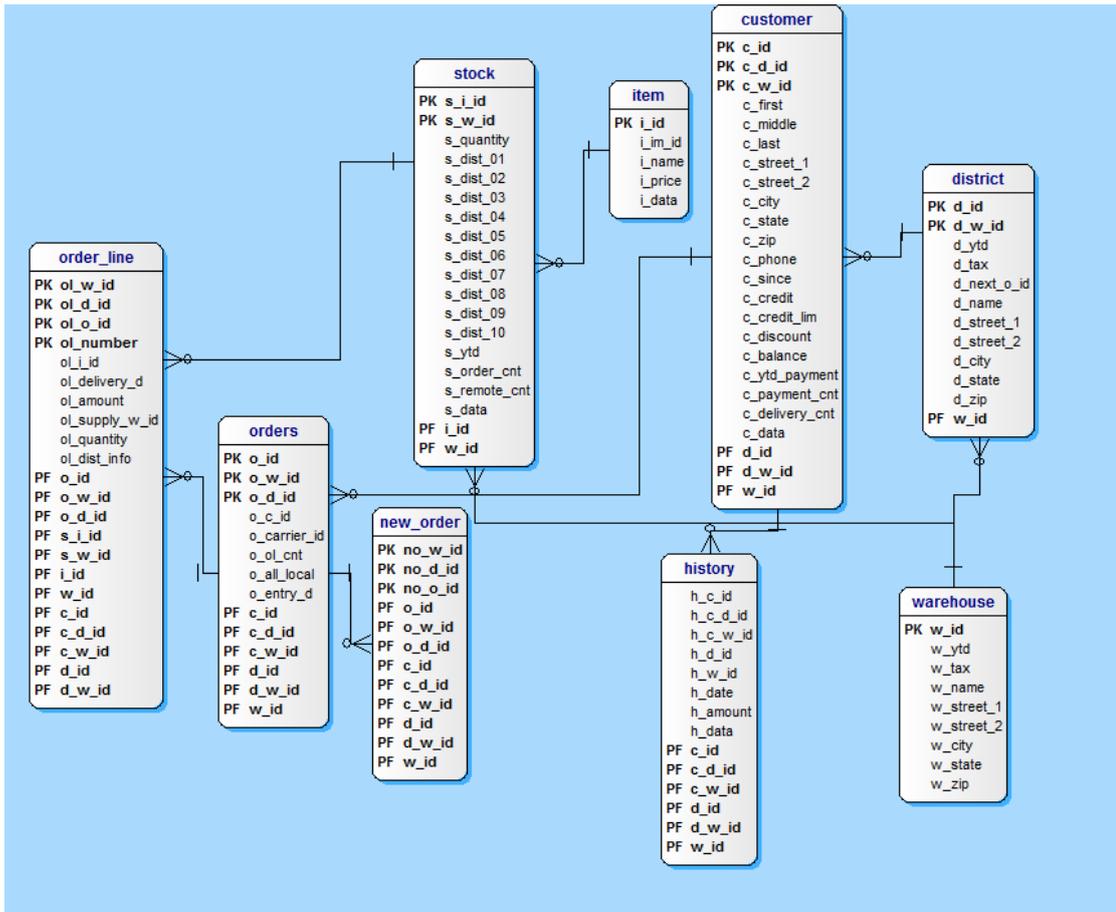


Figura III.15: Esquema de Base de datos para las pruebas

Fuente: <http://hammerdb.com/>

- Descripción de la base de datos

TPC-C implementa un sistema informático para cumplir con los pedidos de los clientes para ofrecer productos de una empresa. La compañía vende 100.000 artículos y mantiene sus existencias en los almacenes. Cada almacén tiene 10 distritos de ventas y cada distrito sirve a 3.000 clientes. Los clientes llaman a la empresa cuyos operadores toman la orden, cada pedido contiene una serie de artículos. Las órdenes son por lo general satisfechos desde el almacén local, sin embargo, un pequeño número de artículos que no están en la acción en un punto determinado en el tiempo son suministrados por un almacén alternativo.

La **figura III.16** ilustra el sistema de cómputo de órdenes que simula la herramienta Hammerdb

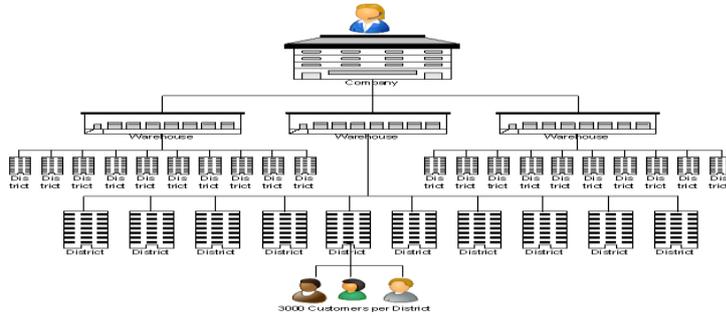


Figura III.16: Sistema de cómputo de órdenes

Fuente: <http://hammerdb.com/>

El script para generar la base de datos que se utilizó en el entorno de pruebas de consumo de recursos y número de órdenes por minuto se encuentra en el **Anexo II**.

Herramienta Hammerdb:

A continuación se detalla los parámetros que se deben de configurar en la herramienta Hammerdb. Se ingresó valores para conectarse al SGBD PostgreSQL

- ❖ Configuración de conexión a la base de datos, tiempo, número de transacciones y número de usuarios.

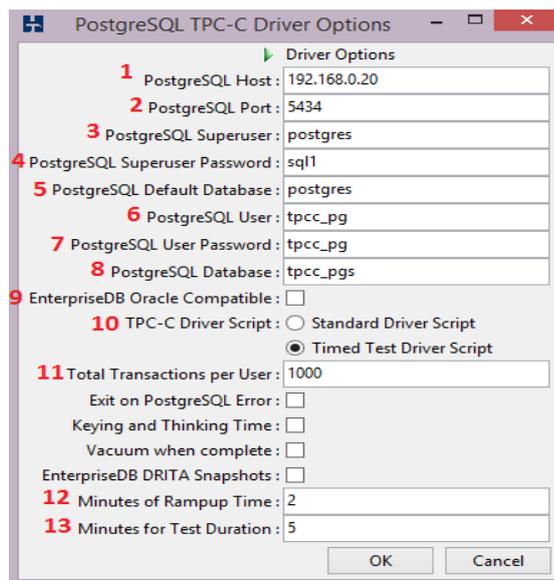


Figura III.17: Parámetros de configuración para TPC-C

Fuente: <http://hammerdb.com/>

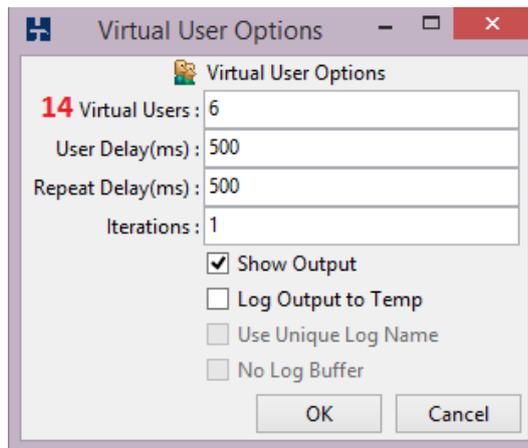


Figura III.18: Configuración de usuarios virtuales para TPC-C

Fuente: <http://hammerdb.com/>

1. **PostgreSQL Host:** nombre del host en donde PostgreSQL se está ejecutando, es el sistema bajo prueba a donde el servidor de generación de carga Hammerdb se conectará.
2. **PostgreSQL Port:** puerto del servicio PostgreSQL, por defeco es el 5432.
3. **PostgreSQL SuperUser:** es el usuario que tiene los suficientes privilegios para crear usuarios (roles) y bases de datos además de la creación del esquema para realizar el test.
4. **PostgreSQL SuperUser Password:** es la contraseña para el super usuario de PostgreSQL que ha sido definido durante la instalación.
5. **PostgreSQL Default Database:** la base de datos por defecto de PostgresSQL es la base de datos que se especifica para la conexión de super usuario. Típicamente esta será "postgres" en una instalación standard de PostgreSQL.
6. **PostgreSQL User:** es el usuario (rol) que será creado y es dueño de la base de datos que contiene el esquema TPC-C que será utilizado para las pruebas.
7. **PostgreSQL User Password:** es la contraseña que será especificada para el usuario de PostgreSQL cuando sea creado.
8. **PostgreSQL Database:** es la base de datos que será creada y poseída por el **PostgreSQL User**, esta base de datos contiene el esquema TPC-C.

- 9. EnterpriseDB Oracle Compatible:** al elegir la opción **EnterpriseDB Oracle Compatible** se crea un esquema usando la característica Oracle compatible de EnterpriseDB en una instalación de PostgreSQL Plus Advanced Server.
- 10. TPC-C Driver Script:** bajo TPC-C Driver script se tiene la opción de elegir entre Standard Driver Script o el Timed Test Driver Script. Esta elección cambiará dinámicamente el Driver Script que está cargado cuando la opción TPC-C Driver Script en el menú ha sido elegida. El Standard Drive Script debería ser elegido cuando se desea crear una carga contra la base de datos y ver el rate de transacciones, pero no desea correr un test de tiempo o desea medir el tiempo del test usted mismo de manera manual. El Standard Drive Script puede ser ejecutado con el Virtual User Output encendido, que desplegará toda la información que cada usuario virtual procesa o al contrario el Virtual User Output apagado para ser capaz de observar solo el rate de transacciones.

En lugar del Standard Drive Script se puede seleccionar el Timed Driver Script. Este produce un número adicional de opciones. Se debería seleccionar el Timed Driver Script cuando se desea ejecutar un test cronometrado. Con el Timed Driver Script el primer usuario virtual hará la sincronización y generará los resultados mientras los usuarios virtuales adicionales ejecutan el workload, además se debe seleccionar el número deseado de usuario +1 cuando se ejecuta el Timed Driver Script.
- 11. Total transactions per User:** Es el número de transacciones que cada usuario virtual procesará antes de cerrar sesión
- 12. Minutes of Rampup Time:** Define el tiempo en minutos que tienen que esperar los usuarios virtuales que ejecutan las pruebas de carga para monitorear la conexión con la base de datos para aumentar la velocidad de las transacciones al almacenar en la cache de la base de datos los datos antes de tomar el primer valor programado e iniciar la prueba.
- 13. Minutes for Test Duration:** Define el tiempo de duración de la prueba, así como el tiempo que el monitor de subprocessos espera después que el primer valor

programado tarda para tomar la segunda señal de que la prueba se haya completado y los usuarios activos completen su carga de trabajo

14. Virtual Users: Cantidad de usuarios que interactuarán de manera simultánea con la base de datos

Ahora para las pruebas de tiempos de respuesta y Tiempo del aseguramiento de la integridad de datos se va a utilizar una base de datos con la misma estructura y tipos de datos en ambos SGBD.

La **figura III.19** ilustra la estructura de base de datos para las pruebas

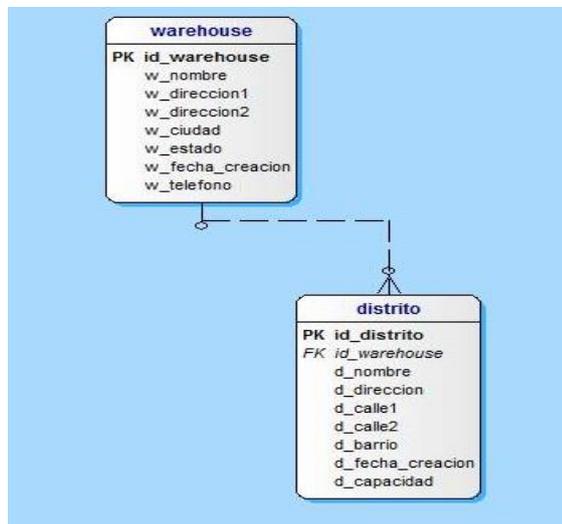


Figura III.19: Esquema 2 de base de datos para las pruebas

Fuente: Johnny Parra, Mario Andrade

El script para generar la base de datos que se utilizó en el entorno de pruebas se encuentra en el **Anexo II**.

3.5 Diseño de las pruebas

Se especifica cada una de las pruebas que se van a implementar, a la vez se definirá sus respectivos pasos y herramientas, lo que permitirá evaluar las tecnologías y posteriormente obtener datos estadísticos de forma clara y concreta.

Cabe recalcar que los escenarios que se van a plantear están enfocados hacia un entorno empresarial, donde se operen grandes volúmenes de datos.

A continuación se detallan cada uno de los escenarios de pruebas.

3.5.1 Prueba 1 - Consumo de recursos

La prueba tendrá como objetivo principal medir el consumo de recursos en el servidor de base de datos durante la ejecución de las pruebas de carga, el valor medido en esta prueba para CPU y memoria RAM será expresado en porcentaje, en cambio el valor medido para la variable RED será evaluada en base a la cantidad de paquetes transmitidos por el enlace.

Las pruebas de carga utilizando la herramienta Hammerdb se ejecutarán en 4 escenarios de acuerdo al número de transacciones con valores de 1000, 10000, 100000, 1000000; en distintos intervalos de tiempo.

Se realizarán 4 pruebas que variarán según del número de usuarios, tiempos de ejecución y número de transacciones. De manera similar se realizará la monitorización del servidor de base de datos utilizando la herramienta Manage Engine.

La **tabla III.XIII** se detalla las características adicionales que se tomarán en cuenta durante la evaluación.

Tabla III.XIII: Prueba 1 – Consumo de recursos

Fuente: Johnny Parra, Mario Andrade

PRUEBA 1 – CONSUMO DE RECURSOS	
VARIABLES	<ul style="list-style-type: none">• Uso de CPU• Uso de memoria RAM• Uso de Red
RECURSOS NECESARIOS	<ul style="list-style-type: none">• Herramienta Hammerdb• Herramienta de Manage Engine• SQL Server 2012 Express• PostgreSQL 9.2.4• Infraestructura de red

3.5.2 Prueba 2 - tiempos de respuesta

Para esta prueba se va hacer uso de una aplicación web desarrollada en el lenguaje php. La prueba tendrá como objetivo principal medir el tiempo de respuesta del servidor de base de datos durante la ejecución de un determinado número de transacciones (inserción y consultas): 1000, 10000, 100000, 1000000

Se utilizará dos pruebas, en la primera se insertarán un determinado número de registros en cada una de las base de datos, la misma que inicialmente tendrá cero registros, posteriormente se procederá a consultar los registros en la base de datos que cuenta con un total de 1000000 de registros que previamente fueron insertados. Para la ejecución de las pruebas antes mencionadas se utilizarán procedimientos almacenados.

A continuación se muestra un procedimiento almacenado que será utilizado para las pruebas. Los demás procedimientos serán adjuntados en el Anexo II.

Procedimiento almacenado para insertar registros en SQL Server

```
Create procedure [dbo].[insertarDistrito_Sql]
As
Declare @i int
Declare @w int
Declare @id int
SET @i = 1
set @w=1
set @id=1
if (Select count(*) from distrito)> 0
    begin
        Set @id= (select max([id_distrito]) from [dbo].[distrito])
        set @id=@id+1
    end
INSERT INTO [dbo].[distrito] ([id_distrito], [id_warehouse], [d_nombre], [d_direccion],
[d_calle1], [d_calle2],[d_barrio], [d_fecha_creacion],[d_capacidad])
VALUES(@id,@w,newid(),newid(),newid(),newid(),newid(), '12/12/2014',rand())
```

Procedimiento almacenado para consultar registros en SQL Server

```
create procedure [dbo].[ConsultarDistritos_Sql]
```

```
@cant int
```

```
As
```

```
    SELECT TOP (@cant)
```

```
        [id_distrito], [w_nombre], [d_nombre], [d_calle1], [d_calle2], [d_fecha_creacion]  
,[d_capacidad] FROM [tdr1_sql].[dbo].[distrito] as D inner join [dbo].[warehouse] as W on  
W.[id_warehouse]=D.[id_warehouse]
```

Para la elaboración y muestra de resultados se utilizó una aplicación web con la siguiente interfaz.

La **figura III.20** ilustra la interfaz de la aplicación web que se utilizó para la presente prueba.



Figura III.20: Interfaz pruebas de tiempo de respuesta

Fuente: Johnny Parra, Mario Andrade

Para la realización de la cada uno de las pruebas, es necesario seleccionar una de las opciones proporcionadas por la interfaz, luego se enviarán los datos mediante el botón enviar y así obtener los resultados de tiempos requeridos. Cabe recalcar que esto aplica para ambos casos

El script php para medir el tiempo de respuesta al insertar y consultar registros se encuentran en el **Anexo II**.

La **tabla III.XIV** se detalla las características adicionales que se tomarán en cuenta durante la evaluación.

Tabla III.XIV: Prueba 2 – Tiempo de respuesta

Fuente: Johnny Parra, Mario Andrade

PRUEBA 2 – TIEMPO DE RESPUESTA	
VARIABLES	<ul style="list-style-type: none">• Tiempo respuesta (segundos)
RECURSOS NECESARIOS	<ul style="list-style-type: none">• Aplicación web en php• SQL Server 2012 Express• PostgreSQL 9.2.4• Infraestructura de red

3.5.3 Prueba 3 - Número de órdenes por minuto

La prueba tendrá como objetivo principal medir el número de órdenes por minuto realizadas en el servidor de base de datos durante un periodo de tiempo determinado y con distinto número de usuarios concurrentes.

Se realizarán 4 pruebas que variarán según del número de usuarios, el tiempo de ejecución es de 30 minutos y número de transacciones es de 100000 para cada prueba.

El parámetro que ha sido tomado en cuenta para esta prueba ha sido el NOPM (número de órdenes por minuto), debido a que este valor es proporcionado por la herramienta Hammerdb.

La **figura III.21** ilustra la Interfaz de salida de las pruebas en TPC-C

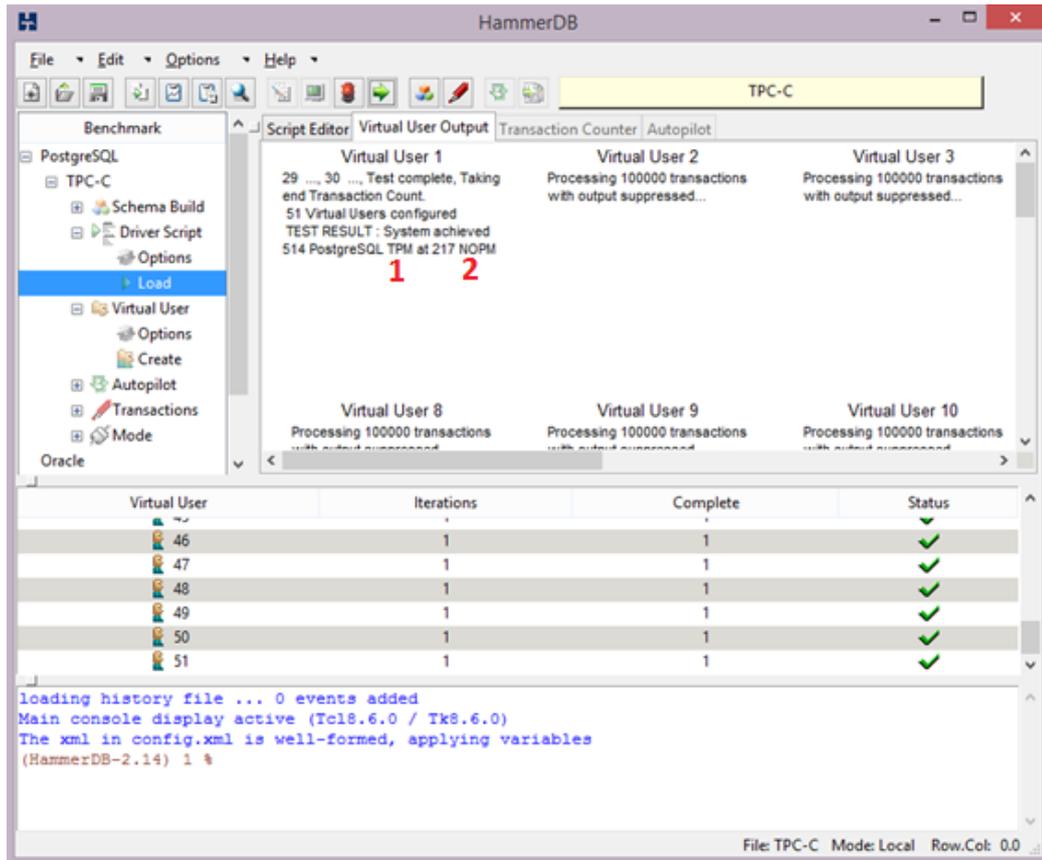


Figura III.21: Interfaz de salida de las pruebas en TPC-C

Fuente: <http://hammerdb.com/>

1. **TPM:** Transacciones por minuto. Este valor no puede ser utilizado para comparar el rendimiento de diferentes SGBD
2. **NOPM:** Viene a ser el número de nuevas órdenes por minuto y es utilizado como una valor estadístico independiente. Este valor si puede ser utilizado para comparar el rendimiento de diferentes SGBD

La **tabla III.XV** se detalla las características adicionales que se tomarán en cuenta durante la evaluación.

Tabla III.XV: Prueba 3 – Número de órdenes por minuto

Fuente: Johnny Parra, Mario Andrade

PRUEBA 3 – NÚMERO DE ÓRDENES POR MINUTO	
VARIABLES	<ul style="list-style-type: none">• Número de órdenes por minuto
RECURSOS NECESARIOS	<ul style="list-style-type: none">• Herramienta Hammerdb• SQL Server 2012 Express• PostgreSQL 9.2

3.5.4 Prueba 4 – Tiempo del aseguramiento de la integridad de datos

Para esta prueba se va hacer uso de una aplicación web desarrollada en el lenguaje de servidor php, mediante la aplicación web se ejecutarán transacciones de inserción hacia la base de datos, la cual cuenta con mecanismos de integridad previamente establecidos para esta prueba, de esta manera se comprobará el tiempo que tarde cada SGBD en aplicar la integridad por cada registro que se vaya a insertar.

Para comprobar la integridad de dominio en la base de datos, se ha creado una restricción **check** para el campo *d_fecha_creacion* de la tabla distrito, el cual valida que la fecha ingresada sea mayor o igual a la fecha actual.

Con el fin de comprobar la integridad definida por el usuario se ha creado un trigger, el cual ingresa en el campo *d_direccion* de la tabla distrito una concatenación de los campos *d_calle1* y *d_calle2* de la misma tabla.

Se comprobará la integridad de entidad al momento de insertar los registros, donde ninguno de ellos debe contener un identificador repetido, en este caso mediante la declaración de un PK en la tabla distrito.

Por último se comprobará la integridad referencial al momento de la inserción de registros, para lo cual el distrito contiene una clave foránea de la tabla warehouse.

Se realizará una prueba donde se insertaran un número de 10000 registros desde la aplicación web, el mismo que invoca al procedimiento almacenado que se encuentra en el SGBD,

A continuación se muestra el trigger y una restricción check que fueron utilizados en la prueba para el SGBD SQL Server. Los demás serán adjuntados en el **Anexo II**.

--restricción check

```
Alter table dbo.districto
add constraint CK_Fecha
check (d_fecha_creacion>= getdate());
go
```

--trigger

```
create trigger dbo.ConcatenarCalles on dbo.districto
after insert
as
begin
update dbo.districto set d_direccion= concat(d_calle1 ,',', d_calle2 )
end
go
```

La **tabla III.XVI** se detalla las características adicionales que se tomarán en cuenta durante la evaluación.

Tabla III.XVI: Prueba 4 - Tiempo del aseguramiento de la integridad de datos

Fuente: Johnny Parra, Mario Andrade

PRUEBA 4 - TIEMPO DEL ASEGURAMIENTO DE LA INTEGRIDAD DE DATOS	
VARIABLES	<ul style="list-style-type: none">• Integridad de Entidad [25].• Integridad de Dominio [25]• Integridad referencial [25]• Integridad definida por el usuario [25]
RECURSOS NECESARIOS	<ul style="list-style-type: none">• Aplicación web en php• SQL Server 2012 Express• PostgreSQL 9.2.4• Infraestructura de red

3.5.5 Métodos para la evaluación de resultados

Se especifica una serie de pasos que servirán para evaluar las tecnologías de forma comparativa, los mismos que permiten mostrar mediante datos estadísticos de forma concreta y clara los resultados obtenidos por cada plataforma.

Luego de una investigación previa en busca de una fórmula que defina lo que es el rendimiento orientado a base de datos, se ha logrado determinar que no existe una fórmula específica aplicable al escenario en estudio. Por consiguiente los autores han definido la fórmula de rendimiento a evaluar en la presente investigación, donde:

- CR.- Consumo de recursos
- NOPM.- Número de órdenes por minuto
- TDR.- Tiempo de respuesta
- IDD.- Tiempo del aseguramiento de la integridad de datos

$$\text{Rendimiento} = \text{CR (25 \%)} + \text{NOPM (25 \%)} + \text{TDR (25 \%)} + \text{IDD (25 \%)}$$

Para la evaluación de cada una de las pruebas se tomará en cuenta una escala de valores especificados en la siguiente **Tabla III.XVII**, con los cuales se podrá valorar de forma porcentual y cualitativa para poder determinar cuál de los SGBD es mejor.

Tabla III.XVII: Valores referenciales para evaluar el rendimiento de los SGBD

Fuente: Johnny Parra, Mario Andrade

Porcentaje (%)	Valor cuantitativo	Valor cualitativo	Descripción
>= 76 y <=100	4	Excelente	Cumple de manera satisfactoria con la mayoría de los requerimientos de las pruebas
>= 51 y <=75	3	Bueno	Cumple con muchos requerimientos de las pruebas
>= 26 y <=50	2	Regular	Cumple con algunos requerimientos de las pruebas
>= 0 y <=25	1	Malo	Cumple con muy pocos requerimientos de las pruebas

Los parámetros a tomarse en cuenta son los siguientes:

- **Consumo de recursos (CR):**

El porcentaje que se ha determinado para evaluar a cada una de las variables referentes al consumo de recursos se ha tomado como referencia de estudios previamente realizados [4] [12] [17].

- CPU = 50 %
- RAM= 30 %
- RED= 20 %

La **Tabla III.XVIII** que se muestra a continuación muestra el rango de valores que servirán de guía para la evaluación de los recursos CPU y RAM.

Producto de una investigación previa se obtuvo como valor referencial que el uso de CPU no debe exceder el 70% de su capacidad [2], el resto de rangos expresados en la **Tabla III.XVIII** han sido propuestos por los autores de este estudio al no haber encontrado una guía que pudiese aplicarse a este caso.

Tabla III.XVIII: Rango de valores para la evaluación de los recursos CPU y RAM

Fuente: Johnny Parra, Mario Andrade

RECURSO	RANGO	VALOR CUANTITATIVO	VALOR CUALITATIVO
CPU	≥ 0 y ≤ 10	4	Excelente
	≥ 11 y ≤ 40	3	Bueno
	≥ 41 y ≤ 70	2	Regular
	≥ 71 y ≤ 100	1	Malo
RAM	≥ 0 y ≤ 25	4	Excelente
	≥ 26 y ≤ 50	3	Bueno
	≥ 51 y ≤ 75	2	Regular
	≥ 76 y ≤ 100	1	Malo

Se propone que para obtener la equivalencia del uso de red puesto que no se encontró un rango o parámetros aceptables como referencia para la investigación que se pudieran aplicar a este estudio en específico, se procederá a asignar un valor proporcional entre el valor máximo y el valor mínimo para cada uno de los escenarios que incluyan este valor, entonces a aquel SGBD que presente valores más óptimos en la medición se le asignara el 20% correspondiente a este parámetro y un valor proporcional para el otro SGBD.

- **Número de órdenes por minuto (NOPM)**

Para el análisis de la presente prueba se tomará como referencia un estudio previo para lo cual se utilizará un valor cuantitativo con un rango del 1 al 10 para evaluar el número de órdenes por minuto realizadas por cada SGBD, entonces se asignará el máximo valor de 10 para el SGBD que ejecute el mayor número de transacciones y se asignará un valor proporcional para el otro SGBD [36].

Luego de este análisis, con los datos obtenidos se obtendrá una valoración sobre el 100%, al mismo tiempo que se calificará de forma cualitativa la presente prueba, para finalmente asignarle un valor proporcional de 25 %, el cual ha sido establecido para esta prueba.

- **Tiempo de Respuesta (TDR)**

Para el análisis de esta prueba se tomará en cuenta los porcentajes proporcionados en la **Tabla III.XIX**, dichos valores han sido propuestos por los autores de este estudio, debido a que no existe un estudio previo donde se defina el nivel importancia los tipos de transacciones en un SGBD.

Tabla III.XIX: Valores referenciales para evaluar el tiempo de respuesta

Fuente: Johnny Parra, Mario Andrade

TIPO TRANSACCIÓN	PORCENTAJE
Inserción	40 %
Consultas	60 %
Total	100 %

Ahora para poder obtener los porcentajes mencionados en la **Tabla III.XIX** se tomará como referencia un estudio previo y se procederá a asignar un valor cuantitativo de un rango del 1 al 10 para evaluar el tiempo que se demore cada inserción o consulta, donde se tomará como referencia el máximo valor de 10 para el SGBD que ejecute la prueba en menor tiempo y se tomará un valor proporcional para el otro SGBD, de esta manera se podrá evaluar cual SGBD realiza las transacciones propuestas en menores periodos de tiempo **[36]**.

Luego de este análisis, con los datos obtenidos se obtendrá una valoración sobre el 100%, al mismo tiempo que se calificará de forma cualitativa la presente prueba, para finalmente asignarle un valor proporcional de 25 %, el cual ha sido establecido para esta prueba.

- **Tiempo del aseguramiento de la integridad de datos (IDD)**

Dado que no se encontró un rango o parámetros aceptables como referencia para la investigación que se pudieran aplicar a este estudio en específico, se propone que para obtener la equivalencia al evaluar la Integridad se procederá a asignar un valor proporcional entre el valor máximo y el valor mínimo para cada uno de los escenarios que incluyan este valor, entonces a aquel SGBD que presente valores más óptimos en la medición se le asignara el 25% correspondiente a este parámetro **[36]**.

Luego de este análisis, con los datos obtenidos se obtendrá una valoración sobre el 100%, al mismo tiempo que se calificará de forma cualitativa la presente prueba, para finalmente asignarle un valor proporcional de 25 %, el cual ha sido establecido para esta prueba.

CAPÍTULO IV

4. ANÁLISIS COMPARATIVO ENTRE LOS SGBD POSTGRESQL Y SQL SERVER

4.1 Ejecución de las pruebas y tabulación de resultados

4.1.1 Prueba 1 - consumo de recursos

Para la prueba de consumo de recursos se ejecutaron 4 pruebas distintas variando el número de transacciones, el número de usuarios y el tiempo de duración, en la **Tabla IV.XX** se especifican los valores utilizados en cada prueba.

Tabla IV.XX: Pruebas para consumo de recursos

Fuente: Johnny Parra, Mario Andrade

Nº Prueba	NÚMERO DE USUARIOS	DURACIÓN DE LA PRUEBA (min)	NÚMERO DE TRANSACCIONES POR USUARIO
1	5	5	1000
2	5	50	10000
3	5	240	100000
4	100	480	1000000

- **USO DE CPU**

Las cantidades especificadas en la **Tabla IV.XXI** representan el porcentaje de utilización de CPU sobre el 100% durante la ejecución de las pruebas en los SGBD PostgreSQL y SQL Server.

Tabla IV.XXI: Resultados de la prueba uso de CPU

Fuente: Johnny Parra, Mario Andrade

Uso de CPU				
CANTIDAD DE TRANSACCIONES	TIEMPO (min)	Nº USUARIOS	POSTGRES %	SQLSERVER %
1000	5	5	6,71%	10,87%
10000	50	5	23,55%	8,89%
100000	240	5	11,46%	15,45%
1000000	480	100	22,28%	15,51%
PROMEDIO			16%	12,68%

En la **Figura IV.22** se puede evidenciar los porcentajes obtenidos sobre el 100% de cada uno de los SGBD en las pruebas planteadas sobre el uso de CPU.

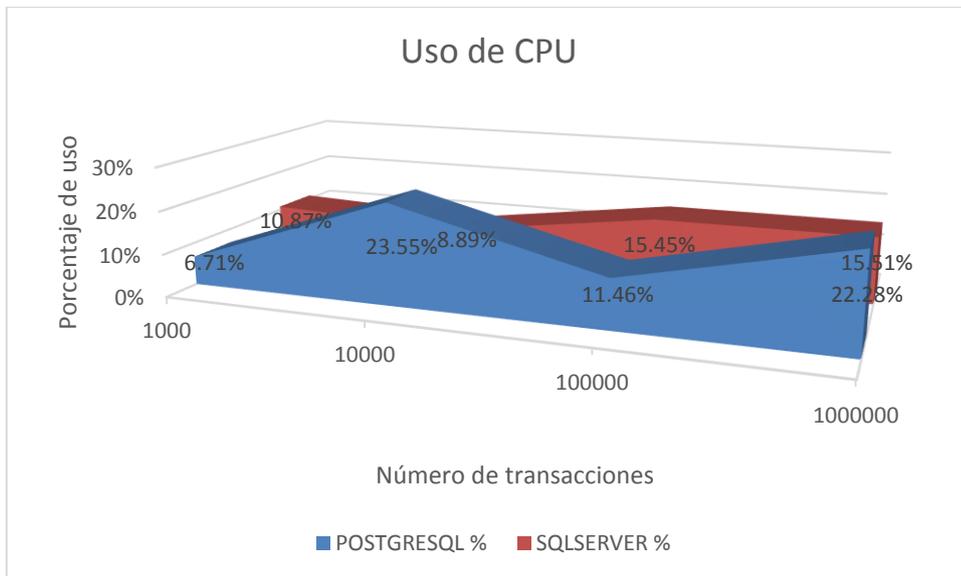


Figura IV.22: Porcentajes de la prueba uso de CPU

Fuente: Johnny Parra, Mario Andrade

En la **Figura IV.23** se puede evidenciar el promedio de los porcentajes obtenidos de cada uno de los SGBD en las pruebas planteadas en la prueba uso de CPU sobre el 100%.

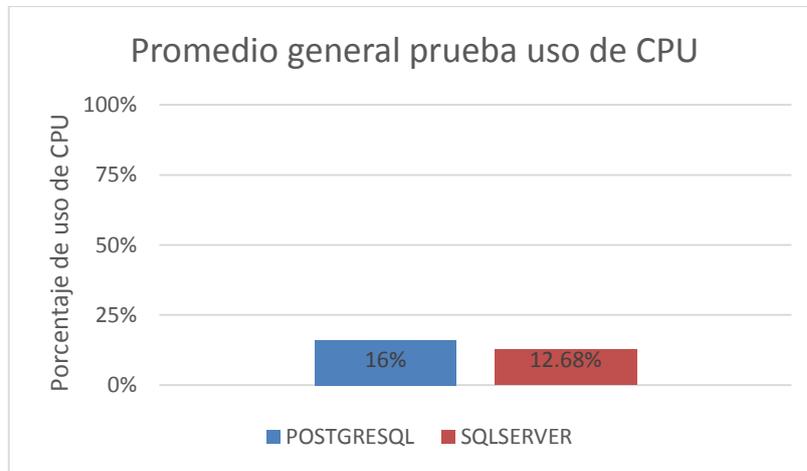


Figura IV.23: Promedio general prueba uso de CPU

Fuente: Johnny Parra, Mario Andrade

De acuerdo al rango de valores para la evaluación del uso del recurso CPU especificado en la **Tabla III.XVIII** de la sección 3.6.5 se puede evidenciar en la **figura IV.23** que no existe una diferencia significativa entre los valores de uso de CPU para los SGBD PostgreSQL que obtuvo un promedio de uso de CPU de 16% y SQL Server que obtuvo un promedio de uso de CPU de 12,68%, entonces puesto que ambos valores caen en el rango que esta entre 11% y 40%, el valor cuantitativo es de 3 y su valor cualitativo corresponde a Bueno en ambos SGBD.

- **USO DE RAM**

Las cantidades especificadas en la **Tabla IV.XXII** representa el porcentaje sobre el 100% de utilización de la RAM durante la ejecución de la prueba de los SGBD PostgreSQL y SQL Server.

Tabla IV.XXII: Resultados de la prueba uso de memoria RAM

Fuente: Johnny Parra, Mario Andrade

Uso de RAM				
CANTIDAD DE TRANSACCIONES	TIEMPO (min)	Nº USUARIOS	POSTGRES %	SQLSERVER %
1000	5	5	30 %	36.56 %
10000	50	5	30.64 %	39.18 %
100000	240	5	38.48 %	48.47 %
1000000	480	100	44.77 %	56.73 %
PROMEDIO			35.97 %	45.24 %

En la **Figura IV.24** se puede evidenciar los porcentajes obtenidos de cada uno de los SGBD en las pruebas planteadas en la prueba uso de RAM sobre el 100%.

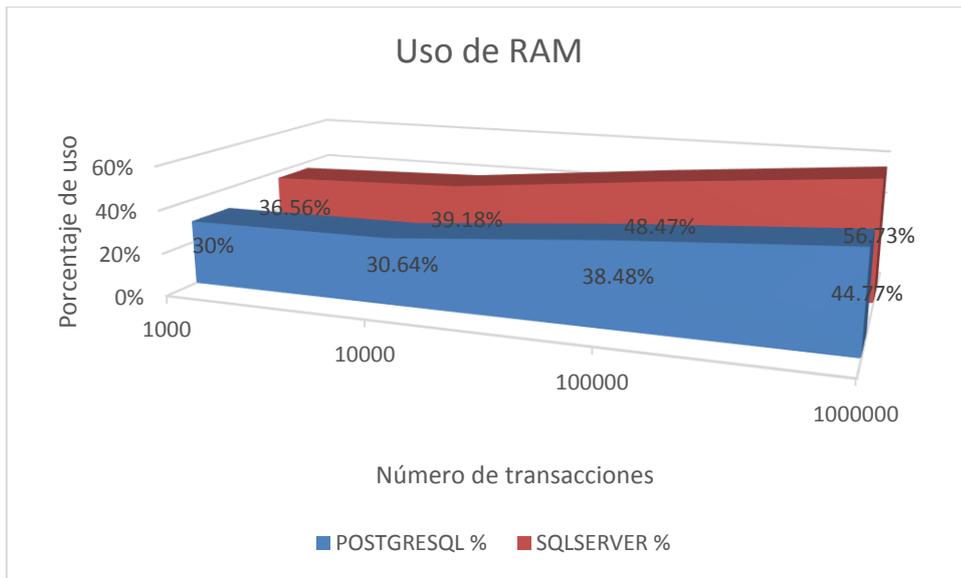


Figura IV.24: Porcentajes de la prueba uso de memoria RAM

Fuente: Johnny Parra, Mario Andrade

En la **Figura IV.25** se puede evidenciar el promedio de los porcentajes obtenidos de cada uno de los SGBD en las pruebas planteadas en la prueba uso de memoria RAM.

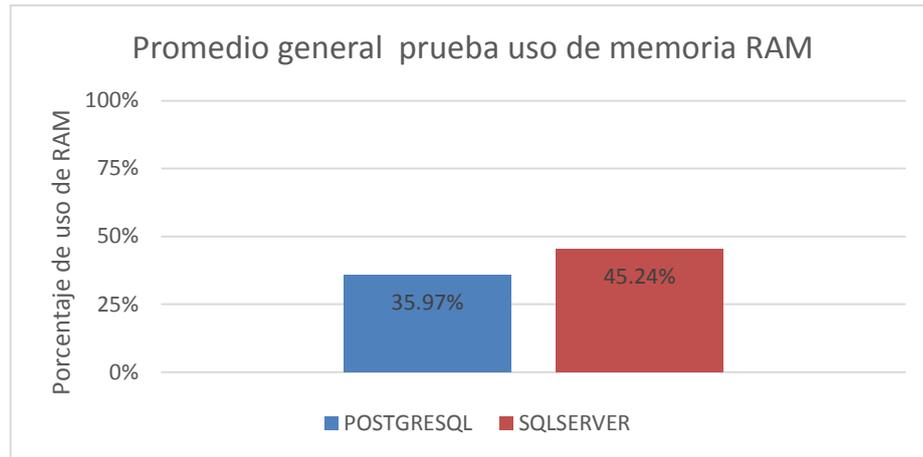


Figura IV.25: Promedio general de la prueba uso de RAM

Fuente: Johnny Parra, Mario Andrade

De acuerdo al rango de valores para la evaluación del uso de recurso memoria RAM especificado en la **Tabla III.XVIII** de la sección 3.6.5, se puede apreciar en la **figura IV.25** que no existe una diferencia significativa entre los valores de uso de RAM para los SGBD PostgreSQL que obtuvo un promedio de uso de memoria RAM de 35,97% y SQL Server que obtuvo un promedio de uso de memoria RAM de 45,24%, entonces ambos valores caen en el rango que esta entre 26% y 50%, el valor cuantitativo es de 3 y su valor cualitativo corresponde a Bueno en ambos SGBD.

- **USO DE RED**

Las cantidades especificadas en la **Tabla IV.XXIII** representan el número de paquetes que se han recibido en la interfaz de red durante la ejecución de la prueba.

Tabla IV.XXIII: Resultados de la prueba uso de RED

Fuente: Johnny Parra, Mario Andrade

Uso de RED				
CANTIDAD DE TRANSACCIONES	TIEMPO (min)	Nº USUARIOS	POSTGRES (Nº paquetes)	SQLSERVER (Nº paquetes)
1000	5	5	700	7000
10000	50	5	15000	22250
100000	240	5	1900	66000
1000000	480	100	3500	30250
PROMEDIO			5275	31375

En la tabla IV.XXIII se puede apreciar que el SGBD SQL Server procesó en promedio la cantidad de 26100 paquetes más que el SGBD PostgreSQL, durante la ejecución de las pruebas, de esta manera se puede concluir que el SGBD SQL Server procesa de manera más rápida las transacciones.

En la **Figura IV.26** se puede evidenciar el número de paquetes enviados por cada uno de los SGBD en las pruebas planteadas en la prueba uso de RED.

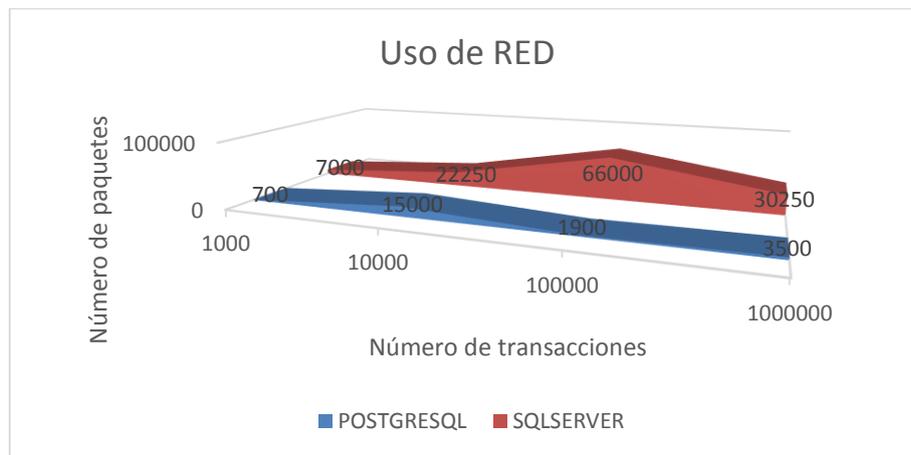


Figura IV.26: Cantidad de paquetes recibidos en la prueba uso de RED

Fuente: Johnny Parra, Mario Andrade

En la **Figura IV.27** se puede evidenciar el porcentaje de paquetes procesados por cada uno de los SGBD en las pruebas planteadas en la prueba uso de RED sobre el 100%.

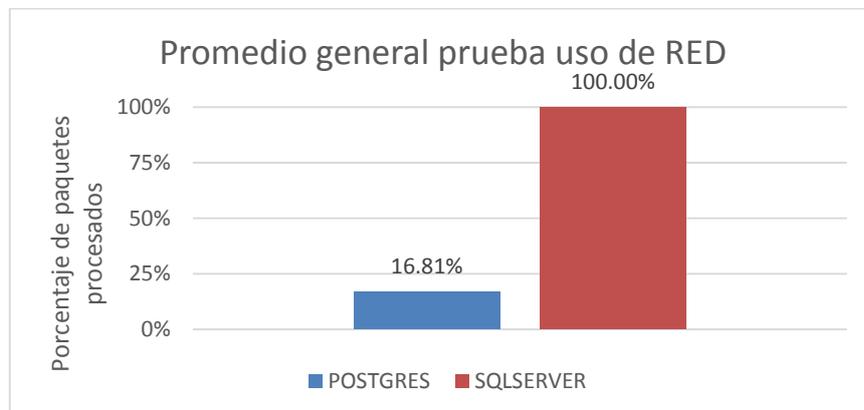


Figura IV.27: Promedio general de la prueba uso de RED

Fuente: Johnny Parra, Mario Andrade

De acuerdo al rango de valores para la evaluación del uso del recurso de RED especificado en la **Tabla III.XVII** de la sección 3.6.5, se puede evidenciar en la **figura IV.27** que existe una diferencia significativa entre los valores de uso de RED para los SGBD PostgreSQL que obtuvo un promedio de uso de RED de 16,81% y SQL Server que obtuvo un promedio de uso de RED de 100%, entonces PostgreSQL cae en el rango de 0% y 25%, el valor cuantitativo es de 1 y su valor cualitativo es Malo, en cambio SQL Server cae en el rango de 76% y 100%, el valor cuantitativo es de 4 y su valor cualitativo es excelente; se puede apreciar que existe una diferencia del 83,19% de SQL Server sobre PostgreSQL.

- **CÁLCULO DE RESULTADOS DE LA PRUEBA**

Para obtener el resultado total de la prueba correspondiente a consumo de recursos se procedió a sumar los valores obtenidos de cada uno de los recursos: Uso de CPU, uso de memoria RAM y Uso de RED.

En la **tabla IV.XXIV** se presenta los porcentajes totales sobre el 100%, obtenido luego de haber realizado las respectivas pruebas; de igual manera sus valores ponderados, los mismos que previamente fueron especificados en la **sección 3.6.5**, de ésta manera se aplicó una regla de tres sobre los porcentajes totales de cada recurso.

Tabla IV.XXIV: Resultado de los porcentajes ponderados de la prueba consumo de recursos

Fuente: Johnny Parra, Mario Andrade

SGBD	Uso de CPU		Uso de RAM		Uso de RED	
	Porcentaje sobre 100%	Ponderado al 50 %	Porcentaje sobre 100%	Ponderado al 30 %	Porcentaje sobre 100%	Ponderado al 20 %
POSTGRES	16 %	42 %	35,97 %	19,21 %	16,81 %	3.36 %
SQLSERVER	12,68 %	43,66 %	45,24 %	16,43 %	100 %	20 %

En la tabla **IV.XXV** se presenta la sumatoria de los porcentajes ponderados de cada recurso, al mismo tiempo que se asignó su valor cualitativo correspondiente al total obtenido por la prueba.

Tabla IV.XXV: Resultado de los porcentajes de la prueba consumo de recursos

Fuente: Johnny Parra, Mario Andrade

Promedio general de la prueba consumo de recursos		
PARÁMETROS	POSTGRESQL	SQL SERVER
Uso de CPU	42 %	43,66 %
Uso de RAM	19,21 %	16,43 %
Uso de RED	3,36 %	20 %
Total	64,57 %	80,09 %
Valor cualitativo	Bueno	Excelente

La **figura IV.28** representa el porcentaje total de la prueba de consumo de recursos sobre el 100%.

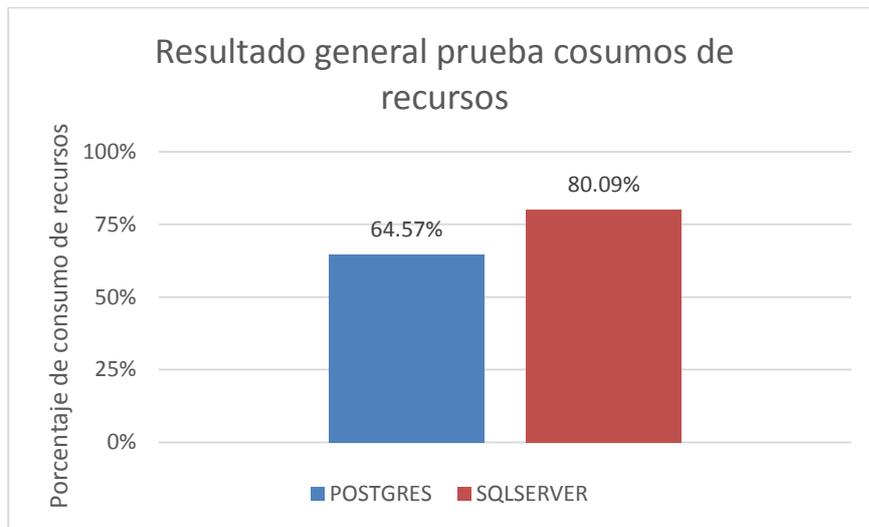


Figura IV.28: Resultado general de la prueba consumo de recursos

Fuente: Johnny Parra, Mario Andrade

En base a los resultados obtenidos se procede a evaluar de forma cualitativa esta prueba, donde el SGBD SQL Server obtiene una calificación de excelente al tener un valor porcentual del 80,09%; mientras que PostgreSQL se lo califica como bueno al haber alcanzado un valor

porcentual del 64,57%. Por lo tanto existe una diferencia significativa de 15,52% del SGBD SQL Server sobre PostgreSQL.

4.1.2 Prueba 2 - Tiempo de respuesta

Las pruebas para medir el tiempo de respuesta de los SGBD se ejecutaron con 2 tipos de operaciones que son: inserción y consulta en ambos casos se varió tanto la cantidad de registros insertados, así mismo se varió la cantidad de registros consultados, el tiempo de cada prueba finalizó al culminar la ejecución de la operación.

La tabla IV.XXVI especifica los valores utilizados en cada escenario.

Tabla IV.XXVI: Escenarios de la prueba tiempo de respuesta

Fuente: Johnny Parra, Mario Andrade

OPERACIONES	NÚMERO DE REGISTROS INSERTADOS/CONSULTADOS	NÚMERO DE USUARIOS
Inserción y consulta de registros	1000	1
	10000	1
	100000	1
	1000000	1

Para la determinación de valores en la prueba tiempo de respuesta se procedió a realizar una regla de 3 con cada uno de los valores medidos, de esta manera al menor tiempo registrado entre los resultados de los dos SGBD se le asignó la máxima puntuación que es de 10, de igual manera para el otro SGBD se calculó su valor proporcional.

A continuación se detalla el procedimiento que se utilizó para las inserciones y consultas efectuadas.

Datos:

- Tiempo de respuesta en PostgreSQL para insertar 1000 registros = 2,87 segundos
- Tiempo de respuesta en SQL Server para insertar 1000 registros = 2,68 segundos

Procedimiento: se obtuvo la equivalencia para el SGBD que registró el mayor tiempo de respuesta al ejecutar la prueba, para lo cual se realizó una regla de 3.

2,68 es a 10, como 2,87 es a **X**

Cálculo

$$(2,68 * 10) / 2,87$$

Resultado

EL SGBD PostgreSQL que registró un tiempo de respuesta de 2,87 segundos para insertar 1000 registros luego de realizado el cálculo obtuvo una equivalencia de 9,87.

De esta manera se procederá a calcular para los demás datos que se necesita para esta prueba

- **INSERCCIÓN DE REGISTROS**

La **Tabla IV.XXVII** muestra los resultados obtenidos de la prueba de tiempo de respuesta y el promedio de los tiempos de respuesta respecto a la inserción de registros, la medición se la realizó en segundos y se varió la cantidad de registros en ambos SGBD.

Tabla IV.XXVII: Resultado total de la prueba TDR, inserción de registros

Fuente: Johnny Parra, Mario Andrade

Inserción de registros				
CANTIDAD REGISTROS	POSTGRESQL		SQL SERVER	
	Tiempo respuesta (s)	Valoración sobre 10	Tiempo respuesta (s)	Valoración Sobre 10
1000	2,87	9,34	2,68	10
10000	49,68	5,24	26,05	10
100000	1078,69	2,37	255,84	10
1000000	>a 28800	0,89	2560,46	10
	Promedio	4,46	Promedio	10

En la **Figura IV.29** se puede evidenciar los tiempos de respuesta registrados en las 4 pruebas de inserción de los SGBD PostgreSQL y SQL Server.

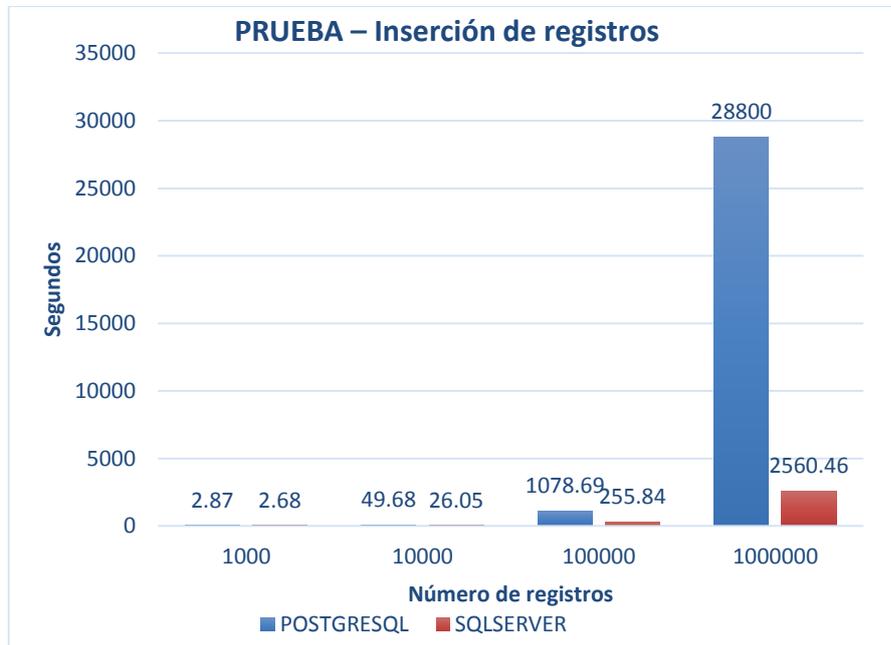


Figura IV.29: Resultados de la prueba 2 (TDR), inserción de registros

Fuente: Johnny Parra, Mario Andrade

En la **Figura IV.29** se evidencia que los tiempos de respuesta de las 4 pruebas de inserción que se realizaron fueron menores en el SGBD SQL Server en comparación con el SGBD PostgreSQL, Cabe resaltar que el SGBD PostgreSQL en la prueba de inserción de 1000000 de registros fue detenida a los 28800 segundos de transcurrida la misma, debido a que era incierto el tiempo en que la prueba iba a finalizar.

La **figura IV.30** muestra los valores porcentuales sobre el 100%, en base a los datos obtenidos en la **Tabla IV.XXVII**.

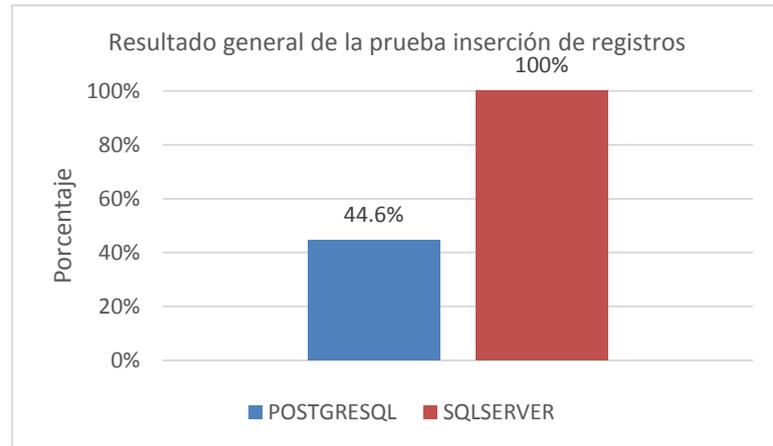


Figura IV.30: Resultado general de la prueba de inserción de registros

Fuente: Johnny Parra, Mario Andrade

De acuerdo al rango de valores para la evaluación de tiempo de respuesta en la **tabla III.XVII** de la sección 3.6.5, se puede apreciar en la **figura IV.30** que existe una diferencia significativa entre los porcentajes de tiempo de respuesta luego de haber ejecutado la inserción de registros para los SGBD PostgreSQL que obtuvo un porcentaje del 44,6% y SQL Server que obtuvo un porcentaje del 100%, de esa manera SQL Server obtuvo un valor cuantitativo de 4 y un valor cualitativo correspondiente a excelente; en cambio PostgreSQL obtuvo un valor cuantitativo de 3 y un valor cualitativo correspondiente a bueno.

- **CONSULTA DE REGISTROS**

La **Tabla IV.XXVIII** muestra los resultados obtenidos de la prueba de tiempo de respuesta y el promedio de los tiempos de respuesta respecto a la consulta de registros, la medición se la realizó en segundos y se varió la cantidad de registros en ambos SGBD.

Tabla IV.XXVIII: Resultado total de la prueba TDR, consulta de registros

Fuente: Johnny Parra, Mario Andrade

Consulta de registros				
CANTIDAD REGISTROS	POSTGRESQL		SQL SERVER	
	Tiempo respuesta (s)	Valoración sobre 10	Tiempo respuesta (s)	Valoración Sobre 10
1000	0,12	10	0,25	4,80
10000	0,46	10	0,61	7,54
100000	2,64	10	6,50	4,06
1000000	32,94	10	64,66	5,09
	Promedio	10	Promedio	5,37

En la **Figura IV.31** se puede evidenciar los tiempos de respuesta registrados en las 4 pruebas de consulta de los SGBD PostgreSQL y SQL Server.

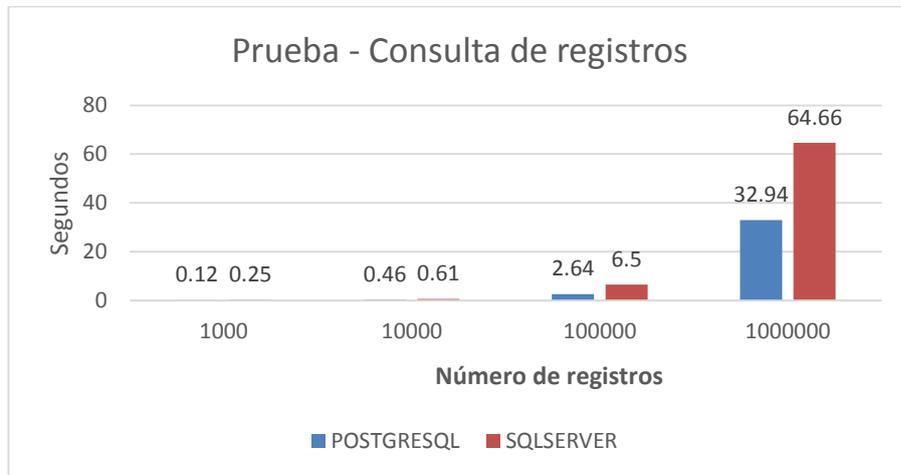


Figura IV.31: Resultados de la prueba 2 (TDR), consulta de registros

Fuente: Johnny Parra, Mario Andrade

En la **figura IV.31** se evidencia que los tiempos de respuesta de las 4 pruebas de consulta que se realizaron fueron mayores en el SGBD SQL Server en comparación con el SGBD PostgreSQL.

La **figura IV.32** muestra los valores porcentuales sobre el 100%, en base a los datos obtenidos en la **Tabla IV.XXVIII**.

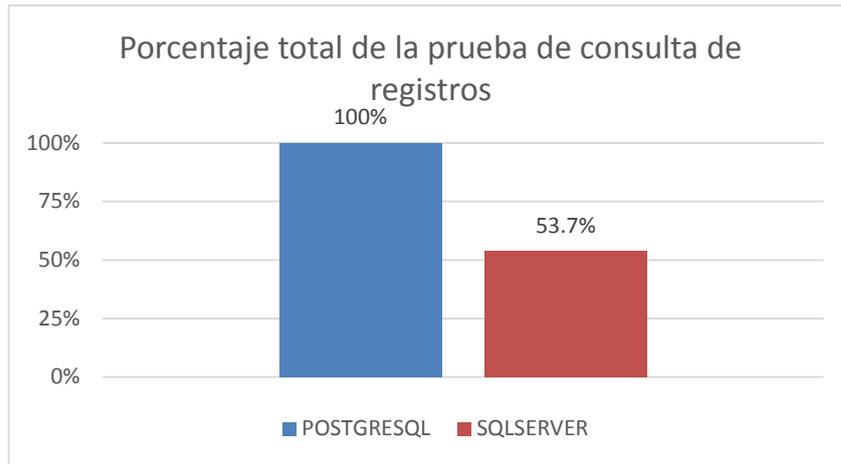


Figura IV.32: Resultado parcial de la prueba 2 Consulta de registros sobre el 100%

Fuente: Johnny Parra, Mario Andrade

De acuerdo al rango de valores para la evaluación de tiempo de respuesta en la **tabla III.XVII** de la sección 3.6.5, se puede apreciar en la **figura IV.32** que existe una diferencia significativa entre los porcentajes de tiempo de respuesta luego de haber ejecutado la consulta de registros; para el SGBD PostgreSQL se obtuvo un porcentaje del 100% y SQL Server que obtuvo un porcentaje del 53,7%, de esa manera SQL Server obtuvo un valor cuantitativo de 3 y un valor cualitativo correspondiente a bueno; en cambio PostgreSQL obtuvo un valor cuantitativo de 4 y un valor cualitativo correspondiente a excelente.

- **CÁLCULO DE RESULTADOS DE LA PRUEBA**

Para obtener el resultado total de la prueba correspondiente a tiempo de respuesta se procedió a sumar los valores obtenidos de las pruebas de: inserción y consultas.

En la **tabla IV.XXIX** se presenta los porcentajes totales sobre el 100%, obtenido luego de haber realizado las respectivas pruebas; de igual manera sus valores ponderados, los mismos que previamente fueron especificados en la **sección 3.6.5**, se aplicaron una regla de tres sobre los porcentajes totales de cada recurso.

Tabla IV.XXIX: Resultado de los porcentajes de la prueba tiempo de respuesta

Fuente: Johnny Parra, Mario Andrade

SGBD	Inserción registros		Consulta registros	
	Porcentaje sobre 100%	Ponderado al 40 %	Porcentaje sobre 100%	Ponderado al 60 %
POSTGRES	44,6 %	17,84 %	100 %	60 %
SQLSERVER	100 %	40 %	53,7 %	32,22 %

En la tabla IV.XXX se presenta la sumatoria de los porcentajes ponderados de inserción y consultas, al mismo tiempo que se asignó su valor cualitativo correspondiente al total obtenido por la prueba.

Tabla IV.XXX: Resultado de los porcentajes de la prueba tiempo de respuesta

Fuente: Johnny Parra, Mario Andrade

Promedio general de la prueba tiempo de respuesta		
PARÁMETROS	POSTGRESQL	SQL SERVER
Inserción registros	17,84 %	40 %
Consulta registros	60 %	32,22 %
Total	77,84 %	72,22 %
Valor cualitativo	Excelente	Bueno

La **figura IV.33** muestra los valores porcentuales de la sumatoria de la pruebas de inserción y consultas. Donde el SGBD que realizó las pruebas en el menor tiempo se hizo acreedor de un mayor porcentaje en una escala de valoración sobre el 100%.

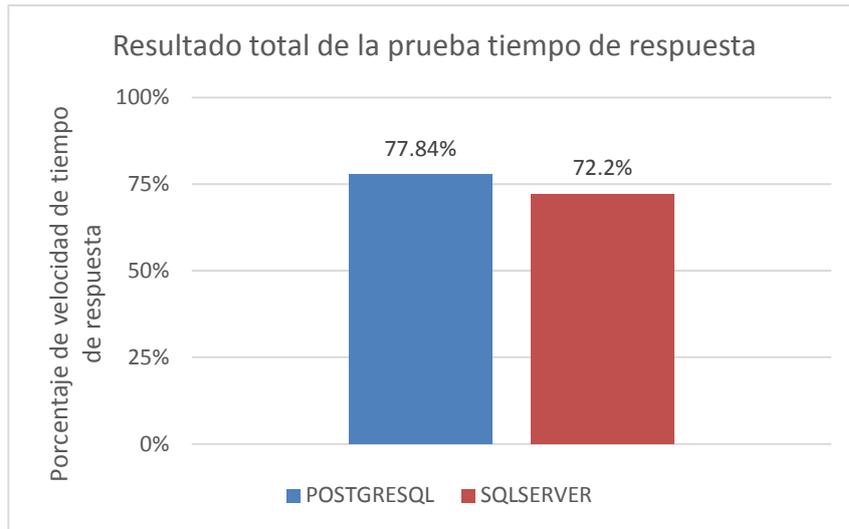


Figura 33: Resultado total de la prueba tiempo de respuesta

Fuente: Johnny Parra, Mario Andrade

Ahora en base a los resultados obtenidos en la **figura IV.33** se procede a evaluar de forma cuantitativa y cualitativa esta prueba basándose en los valores proporcionados en la **tabla III.XVII** de la sección 3.6.5, donde el SGBD SQL Server obtuvo una calificación de 4 y su valor cualitativo corresponde a excelente; mientras que PostgreSQL de forma cuantitativa obtuvo una calificación de 3 y su valor cualitativo corresponde a bueno, por lo tanto existe una diferencia significativa de 5,62% del SGBD SQL Server sobre PostgreSQL.

4.1.3 Prueba 3 - Número de órdenes por minuto

Las pruebas del número de órdenes por minuto se ejecutaron en 5 escenarios distintos variando el número de usuarios, el tiempo asignado para cada escenario fue de 30 minutos y con un número de 100000 transacciones por usuario.

Para la determinación de valores en la prueba de número de órdenes por minuto se procedió a realizar una regla de 3 con cada uno de los valores medidos, de esta manera el SGBD que registró un mayor cantidad de número de órdenes por minuto se le asignó la máxima puntuación que es de 10, de igual manera para el otro SGBD se calculó su valor proporcional.

A continuación se detalla el procedimiento que se utilizó para sacar sus respectivos valores.

Datos:

- Número de órdenes por minuto del SGBD PostgreSQL = 199
- Número de órdenes por minuto del SGBD SQL Server = 1112

Procedimiento: se obtuvo la equivalencia para el SGBD que registró el menor número de transacciones realizadas en un minuto.

$$1112 \text{ es a } 10, \text{ como } 199 \text{ es a } X$$

Cálculo

$$(199 * 10) / 1112$$

Resultado

EL SGBD PostgreSQL que realizó la cantidad de 199 órdenes por minuto luego de realizado el cálculo obtuvo una equivalencia de 1,79.

De esta manera se procederá a calcular para los demás datos que se necesita para esta prueba

La **tabla IV.XXXI** muestra los resultados obtenidos de la prueba de número de órdenes por minuto generados por cada usuario que este interactuando con la base de datos, para esta prueba se varió la cantidad de usuarios.

Tabla IV.XXXI: Resultados de la prueba 3 (NOPM)

Fuente: Johnny Parra, Mario Andrade

Número de órdenes por minuto				
Nº USUARIOS	POSTGRESQL		SQL SERVER	
	Número de órdenes por minuto	Valoración sobre 10	Número de órdenes por minuto	Valoración Sobre 10
100	199	1,79	1112	10
300	119	1,05	1137	10
500	83	0,93	896	10
700	105	1,30	807	10
900	68	0,91	751	10
	Promedio	1,20	Promedio	10

En la **Figura IV.34** se puede evidenciar el número de órdenes por minuto que realizaron los SGBD PostgreSQL y SQL Server en las 5 pruebas efectuadas.

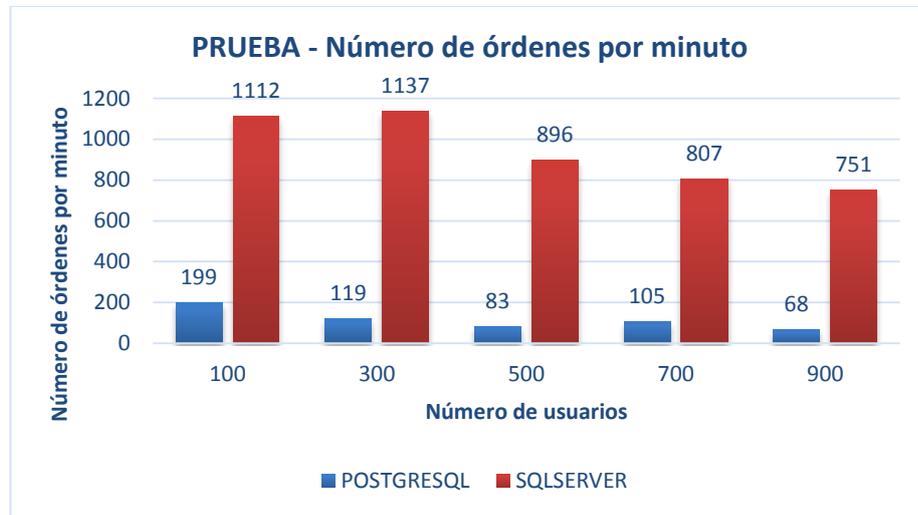


Figura IV.34: Resultados de la prueba 3 (NOPM)

Fuente: Johnny Parra, Mario Andrade

En la **figura IV.35** se presenta valores porcentuales sobre el 100% en base a los datos obtenidos en la **tabla IV.XXXI**, donde el SGBD que logre un mayor valor se considerará como mejor, al haber realizado un mayor número de órdenes por minuto.

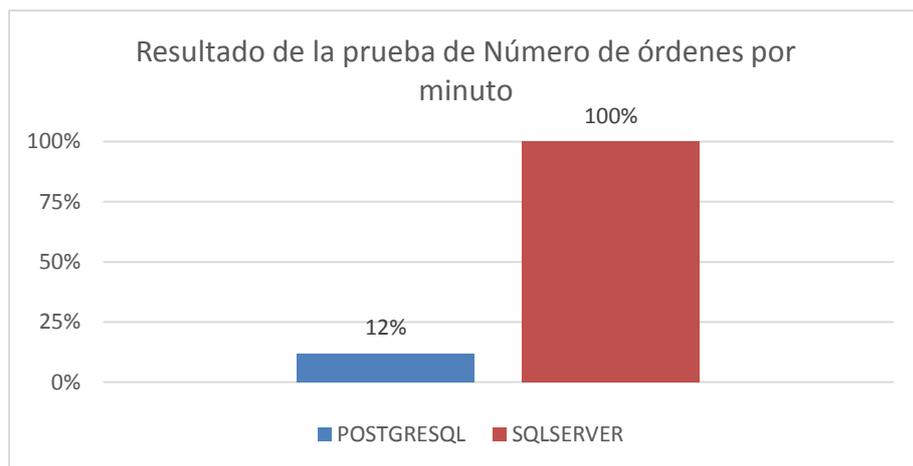


Figura IV.35: Resultado total de la prueba número de órdenes por minuto sobre el 100'

Fuente: Johnny Parra, Mario Andrade

Ahora en base a los resultados obtenidos en la **figura IV.35** se procede a evaluar de forma cuantitativa esta prueba basándose en los valores especificados en la **tabla III.XVII** de la sección 3.6.5, donde el SGBD SQL Server obtuvo una calificación de 4 y su valor cualitativo corresponde a excelente al tener el máximo valor porcentual del 100%; mientras que PostgreSQL de forma cuantitativa obtuvo una calificación de 1 y su valor cualitativo corresponde a malo. Por lo tanto existe una diferencia significativa de 88% del SGBD SQL Server sobre PostgreSQL.

4.1.4 Prueba 4 – Tiempo del aseguramiento de la integridad de datos

Para la presente prueba se midió el tiempo de respuesta que le tomaba a los SGBD en ejecutar la inserción de 10000 registros de esta manera comprobó que tiempo le toma a cada SGBD en validar los diferentes tipos de integridad como: integridad de dominio, integridad de entidad, integridad referencial y la integridad definida por el usuario, la orden fue enviada desde un cliente web hacia el servidor.

Para la determinación de valores en la prueba de Tiempo del aseguramiento de la integridad de datos se procedió a realizar una regla de 3 con cada uno de los valores medidos, de esta manera el SGBD que registró un menor tiempo se le asignó la máxima puntuación que es de 10, de igual manera para el otro SGBD se calculó su valor proporcional.

A continuación se detalla el procedimiento que se utilizó para sacar sus respectivos valores.

Datos:

- Tiempo en verificar integridad SGBD PostgreSQL = 146,46 segundos
- Tiempo en verificar integridad SGBD SQL Server = 4698,72 segundos

Procedimiento: se obtuvo la equivalencia para el SGBD que registró el mayor tiempo en realizar la operación.

146,46 es a 10, como 4698,72 es a **X**

Cálculo

$$(146,46 * 10) / 4698,72$$

Resultado

EL SGBD PostgreSQL que realizó la operación en 4698,72 segundos luego de realizado el cálculo obtuvo una equivalencia de 0,31.

La **tabla IV.XXXII** muestra los resultados obtenidos de la prueba de Tiempo del aseguramiento de la integridad de datos, donde se muestra el tiempo que se demoró en ejecutar la transacción cada SGBD.

Tabla IV.XXXII: Resultados de la prueba de Integridad

Fuente: Johnny Parra, Mario Andrade

Tiempo del aseguramiento de la integridad de datos				
CANTIDAD REGISTROS	POSTGRESQL		SQL SERVER	
	Tiempo (s)	Valoración sobre 10	Tiempo (s)	Valoración Sobre 10
10000	4698,72	0,31	146,46	10
	Promedio	0,31	Promedio	10

La **figura IV.36** representa el tiempo en segundos que le tomó a cada SGBD en validar la integridad de datos, luego de haber insertado una cierta cantidad de registros.

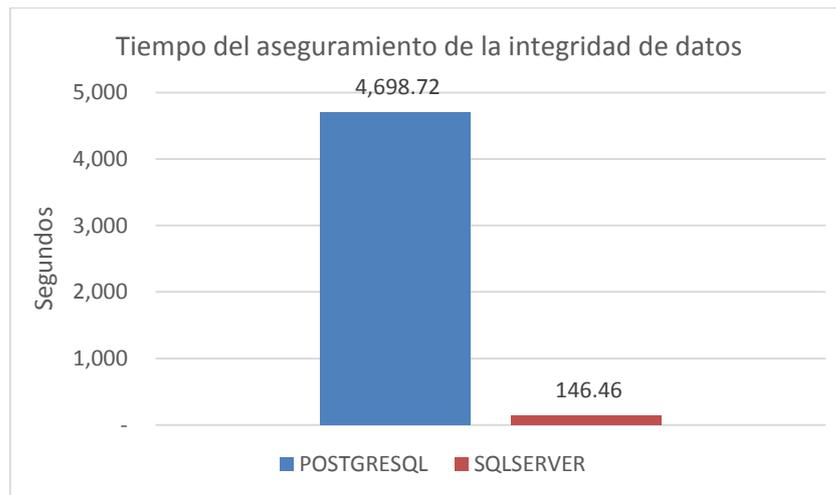


Figura IV.36: Resultado prueba tiempo aseguramiento de la integridad de datos

Fuente: Johnny Parra, Mario Andrade

En la siguiente **figura IV.37** se presenta valores porcentuales sobre el 100% en base a los datos obtenidos en la **tabla IV.XXXII**, donde que el SGBD que logre un mayor valor se considerará como mejor, al haber insertado en menor tiempo la cantidad de registros especificada y aplicado la integridad de datos al mismo tiempo.

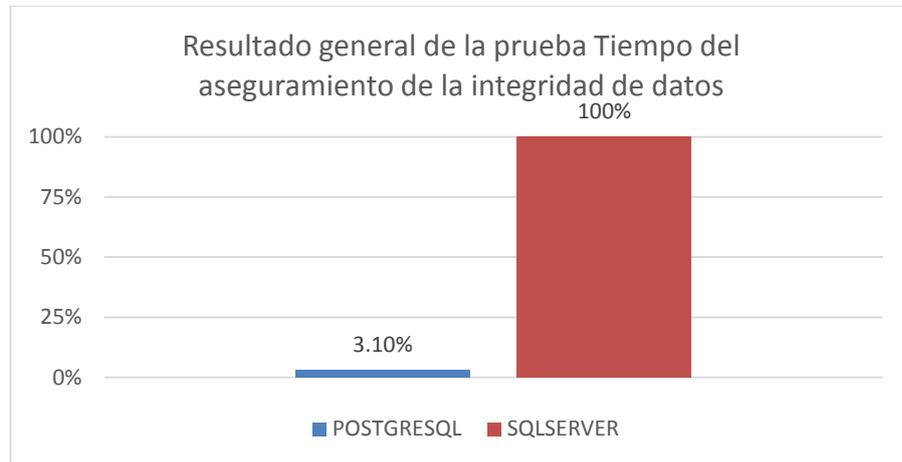


Figura IV.37: Resultado total de la prueba Tiempo del aseguramiento de la integridad de datos

Fuente: Johnny Parra, Mario Andrade

Ahora en base a los resultados obtenidos en la **figura IV.37** se procede a evaluar de forma cuantitativa esta prueba basándose en los valores proporcionados en la **tabla III.XVII** de la sección 3.6.5, donde el SGBD SQL Server obtuvo una calificación de 4 y su valor cualitativo corresponde a excelente al tener el máximo valor porcentual del 100%; mientras que PostgreSQL de forma cuantitativa obtuvo una calificación de 1 y su valor cualitativo corresponde a malo. Por lo tanto existe una diferencia significativa de 96,9% del SGBD SQL Server sobre PostgreSQL.

4.2 Análisis de los datos obtenidos

A continuación se presenta la **Tabla IV.XXXIII**, en la cual se expone los resultados finales de cada una de las pruebas de: consumo de recursos, tiempo de respuesta, número de órdenes por minuto y tiempo del aseguramiento de la integridad de datos, cada uno de ellas ha sido valoradas sobre el 25% de acuerdo a como se definió la fórmula de rendimiento en la **sección 3.6.5**, de esta manera se ha procedido a sumarlas y así obtener un resultado final sobre el 100%, a fin de

poder conocer cuál SGBD ofrece el mejor rendimiento; el SGBD que logre el mayor valor será considerado el mejor.

Tabla IV.XXXIII: Resultado general de las pruebas

Fuente: Johnny Parra, Mario Andrade

Resultado general de las pruebas		
PARÁMETROS	POSTGRESQL	SQLSERVER
Consumo de Recursos	16 %	20 %
Tiempo de respuesta	19,46 %	18,06 %
Número de órdenes por minuto	3 %	25 %
Tiempo del aseguramiento de la integridad de datos	0,78 %	25 %
TOTAL	39,34 %	88,06 %

En base a la **Tabla IV.XXXIII** se presenta la **figura IV.38**, a fin de poder ilustrar los porcentajes alcanzados por cada una de las pruebas en ambos SGBD sobre el 25%.

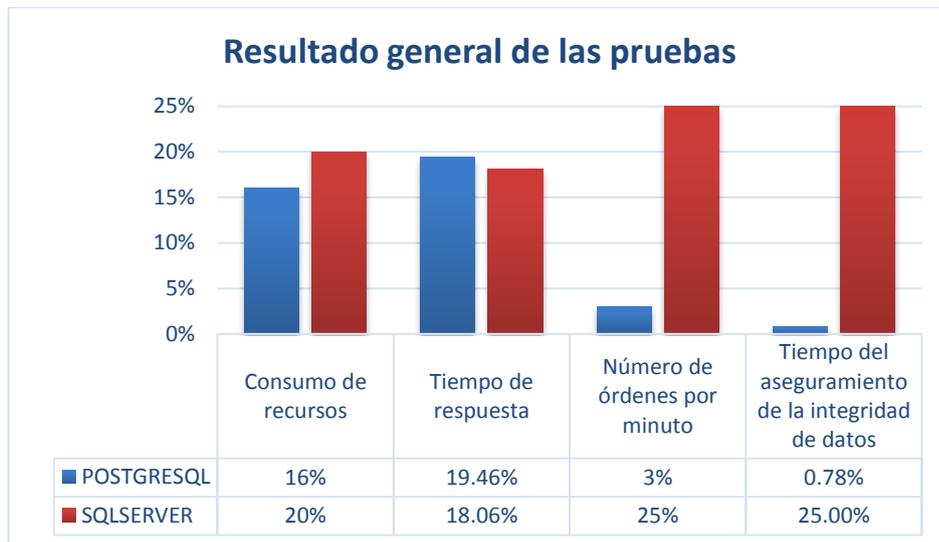


Figura IV.38: Resultados general de las pruebas

Fuente: Johnny Parra, Mario Andrade

La **figura IV.39** detalla el porcentaje final alcanzado por ambos SGBD, luego de haber sumado los resultados de cada una de las pruebas realizadas.

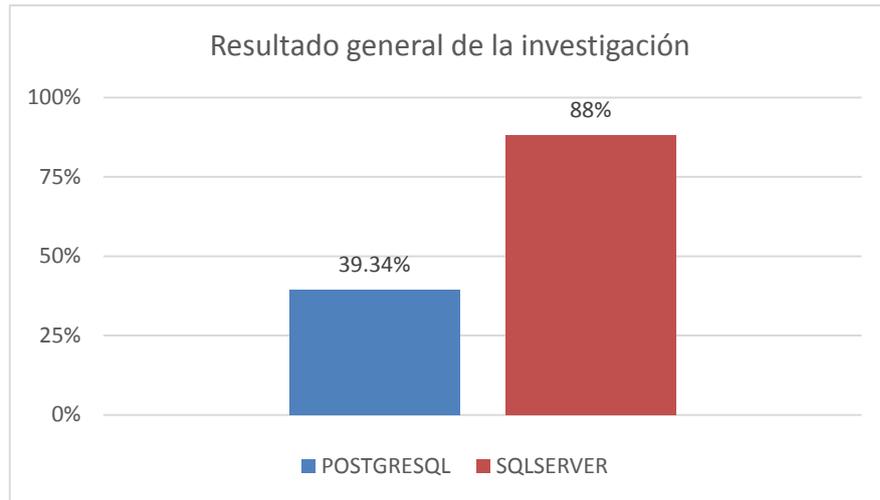


Figura IV.39: Resultado general de la investigación

Fuente: Johnny Parra, Mario Andrade

Ahora en base a los resultados obtenidos en la **figura IV.39** se procede a evaluar de forma cuantitativa esta prueba, donde el SGBD SQL Server obtuvo una calificación de 4 y su valor cualitativo corresponde a excelente al tener un valor porcentual del 88,06%; mientras que PostgreSQL de forma cuantitativa obtuvo una calificación de 2 y su valor cualitativo corresponde a regular. Por lo tanto existe una diferencia significativa de 48,72% del SGBD SQL Server sobre PostgreSQL.

4.3 Comprobación de la hipótesis

4.3.1 Determinación de la población y muestra

Las condiciones para realizar y preparar el escenario de cada una de las pruebas para la presente investigación se detallan a continuación:

Prueba 1 – Consumo de recursos

- El número de usuarios concurrentes fueron de 5 y 100.
- El número de transacciones por usuario fueron de 1000, 10000, 100000 y 1000000 en los siguientes intervalos de tiempo, 5, 50, 240 y 480 minutos respectivamente.

Prueba 2 – Tiempo de respuesta

- El número de transacciones por usuario fueron de 1000, 10000, 100000 y 1000000 únicamente con un usuario.

Prueba 3 – Numero de órdenes por minuto

- El número de usuarios concurrentes fueron de 100, 300, 500, 700 y 900.
- El número de transacciones por usuario fueron de 100000 con una duración de 30 minutos

Prueba 4 – Tiempo del aseguramiento de la integridad de datos

- El número de transacciones por usuario fueron de 10000 únicamente con un usuario.

4.3.2 Planteamiento de la hipótesis

H₀: El SGBD PostgreSQL no ofrece un mejor rendimiento frente al SGBD SQL Server en la implementación de aplicaciones web.

H₁: El SGBD PostgreSQL ofrece un mejor rendimiento frente al SGBD SQL Server en la implementación de aplicaciones web. Leyenda

μ_p = rendimiento del SGBD POSTGRESQL

μ_s = rendimiento del SGBD SQL SERVER

$$H_0: \mu_p < \mu_s$$

$$H_1: \mu_p \geq \mu_s$$

Variable Independiente

- El SGBD POSTGRESQL
- El SGBD SQL SERVER

Variable Dependiente

- El rendimiento

4.3.3 Comprobación de la hipótesis

En base a los resultados de la **figura IV.39** se procede a demostrar la hipótesis planteada.

Debido que el SGBD PostgreSQL no ha obtenido un mejor porcentaje de rendimiento frente al SGBD SQL Server, por lo cual mediante estadística descriptiva se procede a aceptar la hipótesis nula

CAPÍTULO V

5. DESARROLLO DEL SISTEMA ACADÉMICO APLICANDO EL RESULTADO DEL ANÁLISIS A LOS SGBD.

5.1 INFORMACION GENERAL DE LA EMPRESA

5.1.1 Antecedentes

La Escuela De Capacitación Para Conductores Profesionales “CONDUESPOCH” lleva el control de su información de forma semi-automatizada, mediante el uso de hojas electrónicas de Excel para el almacenamiento de información de carácter administrativo-académico y por otra parte se utilizan también hojas de papel que son llenadas a mano, ambas están definidas como instrumentos que proporcionan información a los funcionarios, docentes y estudiantes con respecto a calificaciones, carga horaria, pagos pendientes de estudiantes.

El creciente volumen de información que se genera y gestiona en la institución actualmente, hace que dicha información no este localizada en el momento en que se necesita; esto provoca un consumo adicional e innecesario de tiempo y recursos.

Ante la falta de una herramienta que mantenga la integridad en los datos, evite la duplicación de procesos, que establezca la responsabilidad en cada uno de sus usuarios de acuerdo a sus funciones y mejore la eficiencia de los procesos que actualmente son poco eficientes “CONDUESPOCH” requiere una herramienta que cumpla con características como: usabilidad, mantenibilidad, portabilidad y que resuelva las problemáticas antes mencionadas.

5.1.2 Visión

Ser una institución líder en la formación de conductores profesionales, con ética profesional y valores morales para el desarrollo de la provincia y el país, con calidad, calidez, perseverancia y reconocimiento social.

5.1.3 Misión

Formar conductores profesionales consientes, íntegros, éticos, responsables comprometidos con la sociedad y medio ambiente, a través de la generación, del conocimiento científico y tecnológico, con personal altamente capacitado buscando así contribuir con el mejoramiento de la seguridad vial del país.

5.2 BENEFICIOS DEL SISTEMA

5.2.1 Beneficios tangibles

- Disponibilidad de la información
- Ahorro de recursos
- Menor consumo de tiempo en cada uno de los procesos
- Integridad de datos
- Niveles de seguridad

5.2.2 Beneficios intangibles

- Mejorar el servicio a los usuarios (clientes)
- Mejora en la imagen de la empresa

- Mejor ambiente de trabajo
- Ayuda en la toma de decisiones en la empresa

5.3 FACTIBILIDAD

Previo al desarrollo del proyecto planteado se realizó el estudio de factibilidades con el objetivo de determinar si es viable su implementación, ante esta situación se desarrolló y analizó la factibilidad técnica, factibilidad operativa y factibilidad económica.

5.3.1 Factibilidad Técnica

La empresa CONDUESCPOCH cuenta con el equipamiento necesario para la implementación del proyecto de manera local, con respecto al software para el desarrollo e implementación del sistema se utilizará software libre. Por lo tanto el proyecto es factible técnicamente.

Información más detallada con respecto a los requerimientos de implementación se encuentra adjuntada en el **Anexo III**.

5.3.2 Factibilidad Económica

El costo del presente proyecto asciende a 6732.20 dólares americanos, sin embargo este valor no será desembolsado por la empresa puesto que no cuenta con los recursos económicos suficientes, por ello se ha llegado a un acuerdo mediante el cual no se recibirá ningún emolumento por parte de la empresa a cambio de la flexibilidad en el tiempo establecido de entrega del proyecto; de esta manera proporcionaremos nuestro aporte a la empresa que nos ha abierto sus puertas para que podamos realizar nuestra investigación y aplicar nuestros conocimientos.

Habiendo dicho lo anterior el proyecto es factible económicamente para la empresa, el detalle de los costos calculados se encuentra en el **Anexo III**.

5.3.3 Factibilidad Operativa

Para este apartado se definirá cada una de las personas quienes harán uso del sistema luego de su implementación.

La **tabla V.XXXIV** detalla las personas con sus respectivas funciones, a quienes se brindará una capacitación para el manejo del sistema académico

Tabla V.XXXIV: Resultado general de las pruebas

Fuente: Johnny Parra, Mario Andrade

FUNCIÓN	CAPACITACIÓN
Administrador del Sistema	Administración general de la aplicación Conocimientos básicos de base de datos, ofimática, redes
Inspector	Ingreso de asistencias de los estudiantes
Director/a pedagógico/a	Asignación de materias de docentes Asignación de carga horaria a docentes
Secretaria	Inscripción y matriculación de estudiantes Ingreso de notas materias de grado
Docentes	Ingreso de notas: parciales, suspenso

5.4 ESTIMACIONES

Para el cálculo de las estimaciones se realizó con el modelo matemático Cocomo, las mismas que fueron obtenidos luego de hacer uso del programa USC Cocomo II, en el cual se ingresó cada una de las complejidades de las historias de usuario, cabe recalcar que el tiempo, personal de trabajo y costos son valores nominales los cuales sirven de guía para la proyección del sistema.

Para el desarrollo de este proyecto se dispone de 3 integrantes, 2 los cuales cumplirán el rol de programadores y la otra persona estará encargada de la supervisión y control del desarrollo del sistema.

En base a los resultados obtenidos por el programa, se puede decir que el desarrollo del sistema tendrá una duración aproximada de 2.6 meses y un costo de 27.639,31 dólares americanos.

El detalle de los cálculos realizados en el programa COCOMO II se puede ver en **Anexo IV**.

5.5 ANÁLISIS DE RIESGOS

El análisis de riesgos se lo realizo para evitar futuras complicaciones que puedan interferir en el desarrollo del proyecto. Analizando los riesgos se determinó que el proyecto tiene una baja probabilidad de que se convierta en un problema.

El análisis de riesgo se detallará en el **Anexo V**.

5.6 PLANIFICACIÓN GENERAL DEL PROYECTO

La planificación de este proyecto se basó en la metodología de desarrollo de software SCRUM, se definieron 6 Sprints, en el primer sprint se realizó el estudio de los requerimientos además de la definición y delimitación del alcance del proyecto; en el segundo sprint se definió todo relacionado a la base de datos necesaria para el desarrollo del sistema, también se definió las partes relacionadas a la inscripción, visualización de los estudiantes y la creación de usuarios y roles en el sistema; en el tercer sprint se definió las partes relacionadas al registro y visualización de matrículas, la asignación de materias a docentes y la asignación/visualización de carga horaria a docentes; en el cuarto sprint se definió las partes relacionadas al registro y visualización de asistencias, también el registro y visualización de notas; en el sprint 5 se definió las partes relacionadas a la implementación de seguridad de acceso al sistema y al desarrollo del módulo de administración general del sistema; en el sprint 6 se definió las partes relacionadas a los reportes del sistema, a cada SPRINTS se asignó un tiempo de duración; como fecha de inicio del primer sprint se estableció el 14 junio de 2014.

Los días laborables establecidos para el desarrollo fueron: lunes, martes, miércoles, jueves y viernes.

Las horas laborables se planificaron de la siguiente forma: 8:00 – 12:00 y de 14:00 – 18:00.

Se utilizó el programa Microsoft Project como herramienta para llevar un orden cronológico sobre las actividades dentro del desarrollo del sistema de acuerdo al modelo ágil SCRUM.

La planificación se detalla en el **Anexo VI**.

5.7 ARQUITECTURA DEL SISTEMA

La arquitectura que se va a utilizar para desarrollar el sistema es la arquitectura MVC, el modelo–vista–controlador es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario [11].

La **figura V.40** muestra un esquema simplificado del funcionamiento interno de Symfony2 y cómo se aplican los principios fundamentales de la arquitectura MVC.

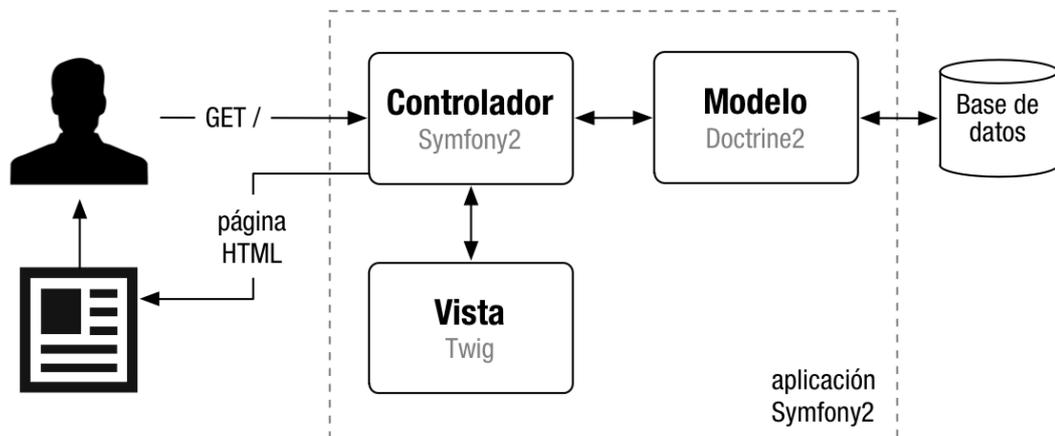


Figura V.40: Arquitectura Modelo Vista Controlador

Fuente: Desarrollo web ágil con Symfony2

El funcionamiento interno siempre es el mismo: 1) el Controlador manda y ordena, 2) el Modelo busca la información que se le pide, 3) la Vista crea páginas con plantillas y datos.

5.7.1 Diagrama de despliegue

El diagrama de despliegue permite mostrar la arquitectura del sistema desde el punto de vista del despliegue (distribución) de los artefactos (elementos determinados del mundo físico que son el resultado de un proceso de desarrollo) del software en los destinos del despliegue.

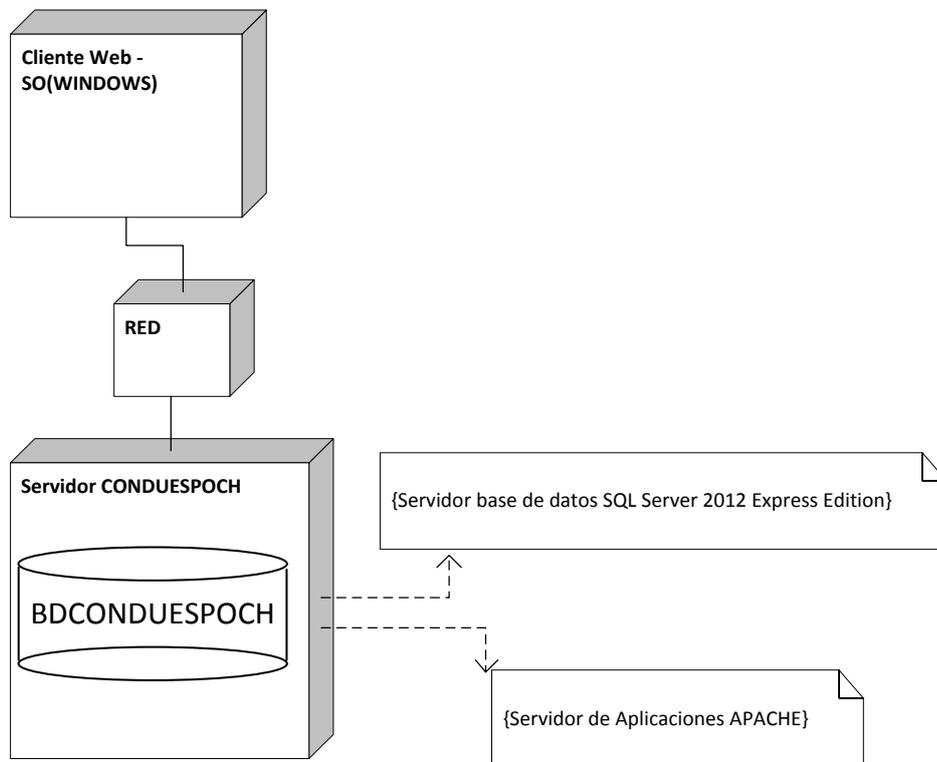


Figura V.41: Diagrama de despliegue

Fuente: Johnny Parra, Mario Andrade

5.7.2 Diagrama de componentes

El diagrama de componentes indica las dependencias lógicas entre componentes software, los mismos que pueden ser fuentes, binarios o ejecutables. Este tipo de diagramas son utilizados para documentar y de la misma manera modelar cualquier arquitectura del software.

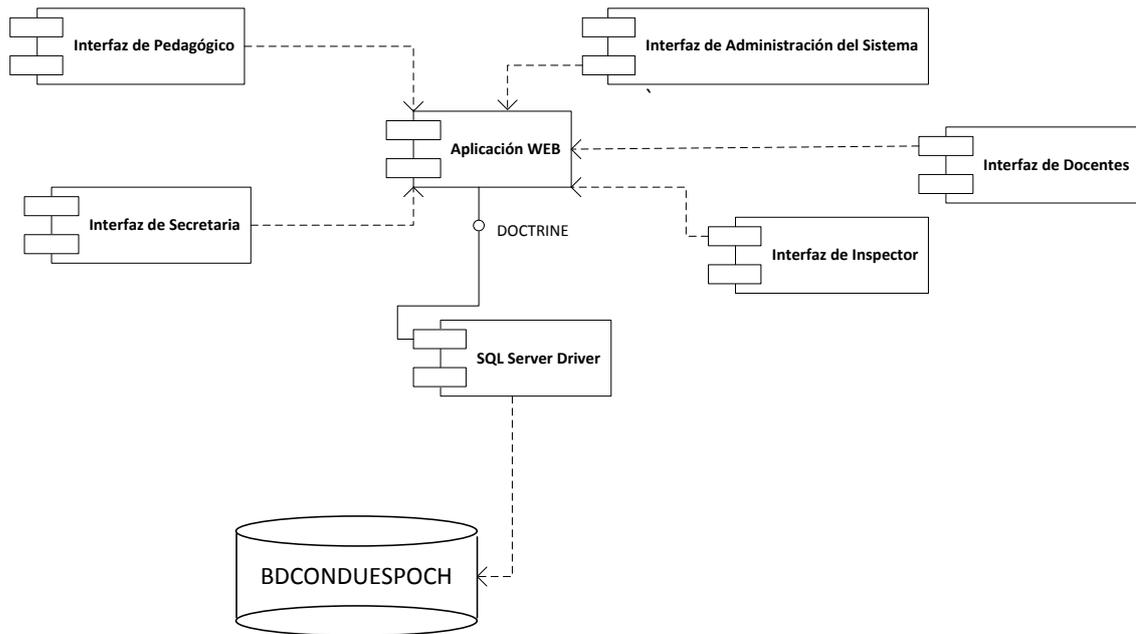


Figura V.42: Diagrama de despliegue

Fuente: Johnny Parra, Mario Andrade

5.8 METODOLOGÍA SCRUM

Es una metodología ágil y flexible para gestionar el desarrollo de software. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación [3].

5.8.1 Características

- Motivación y responsabilidad de los equipos por la auto-gestión, auto-organización y compromiso [3].
- El desarrollo de software se realiza mediante iteraciones, a las que se denomina Sprints [3].
- El resultado de cada sprint es un incremento ejecutable que se muestra al cliente [3].
- Existen reuniones diarias del equipo de desarrollo para coordinación e integración, lo que implica la comunicación verbal directa entre los implicados en el proyecto [3].

5.8.2 Ventajas

- **Buena respuesta frente a cambios:** Scrum acepta que se produzcan cambios en el producto durante el desarrollo del mismo, cualquier nueva característica o modificación es bienvenida al inicio de cada ciclo [3].
- **Aumenta la visibilidad:** Las reuniones mensuales y diarias permiten identificar y corregir problemas casi de inmediato [3].
- **Entrega el mayor valor primero:** Al trabajar sobre una lista priorizada, nos aseguramos de que entregamos primero las características de mayor valor. Y al entregar un producto terminado al final de cada ciclo, no nos arriesgamos a que el proyecto fracase porque no haya dado tiempo a terminarlo [3].
- **Mejora las estimaciones:** Al trabajar en ciclos de duración fija, se aprende rápidamente como de productivo es el equipo y cuántas tareas es capaz de terminar en un tiempo dado [3].
- **Elimina el síndrome del estudiante:** En vez de trabajar de forma relajada al principio del proyecto y acumular estrés los últimos meses, con Scrum el estrés se distribuye al final de los múltiples ciclos de un mes que componen el desarrollo del producto [3].

5.8.3 Roles de SCRUM

En SCRUM existen dos niveles bien diferenciados de implicación en el desarrollo de un proyecto, por un lado están los que están directamente implicados en el desarrollo del mismo, y por otro se encuentra los que solo están afectados de forma tangencial [3].

Dentro de los que están directamente implicados están:

Product Owner o Dueño del producto

Es el interlocutor entre el cliente y el equipo. Cuando está con el equipo él representa la voz del cliente, y cuando está con el cliente él representa la voz del equipo. Es el encargado de indicar qué va a hacerse en cada sprint o que se desea obtener para poder mostrárselo a los interesados. Su labor es asegurarse de que el desarrollo del trabajo se produce dentro de los objetivos que ha marcado el cliente [3].

Sus áreas de responsabilidad son:

- Financiación del proyecto
- Requisitos del sistema
- Retorno de la inversión del proyecto
- Lanzamiento del proyecto

ScrumMaster o Gestor de Scrum

Es un miembro del equipo que desarrolla tareas especiales, su objetivo principal es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. Es el que comunica al Product Owner el desarrollo del sprint, el que se encarga de que se desarrolle normalmente, intenta solucionar los impedimentos que surgen dentro del equipo, en resumen, un representante del equipo que controla que el proceso de un sprint se ejecuta de forma correcta [3].

Sus áreas de responsabilidad son:

- Formación y entrenamiento del proceso
- Incorporación de Scrum en la cultura de la empresa

- Garantía de cumplimiento de roles y responsabilidad

Scrum Team o Equipo

Son los encargados de desarrollar las funcionalidades que el Product Owner ha elegido para cada sprint. Lo ideal es que incluya entre cinco y nueve miembros, y debe ser un equipo auto-gestionado, auto-organizado y multifuncional [3].

Entre los que no están involucrados de forma directa tenemos:

Usuarios

Son el destinatario final del producto, lo prueban y ven si cumple sus expectativas, aportan ideas o necesidades no consideradas [3].

Stakeholders (Clientes, Proveedores, Inversores)

Se refiere a la gente que hace posible el proyecto y para quienes el proyecto producirá el beneficio acordado que lo justifica. Sólo participan directamente durante las revisiones del sprint [3].

5.8.4 Fases

A continuación se ha realizado un esquema con las fases de un proyecto Scrum en el que se van realizando las diferentes 5 fases en forma de ciclos, hasta completar todas las tareas del Backlog.

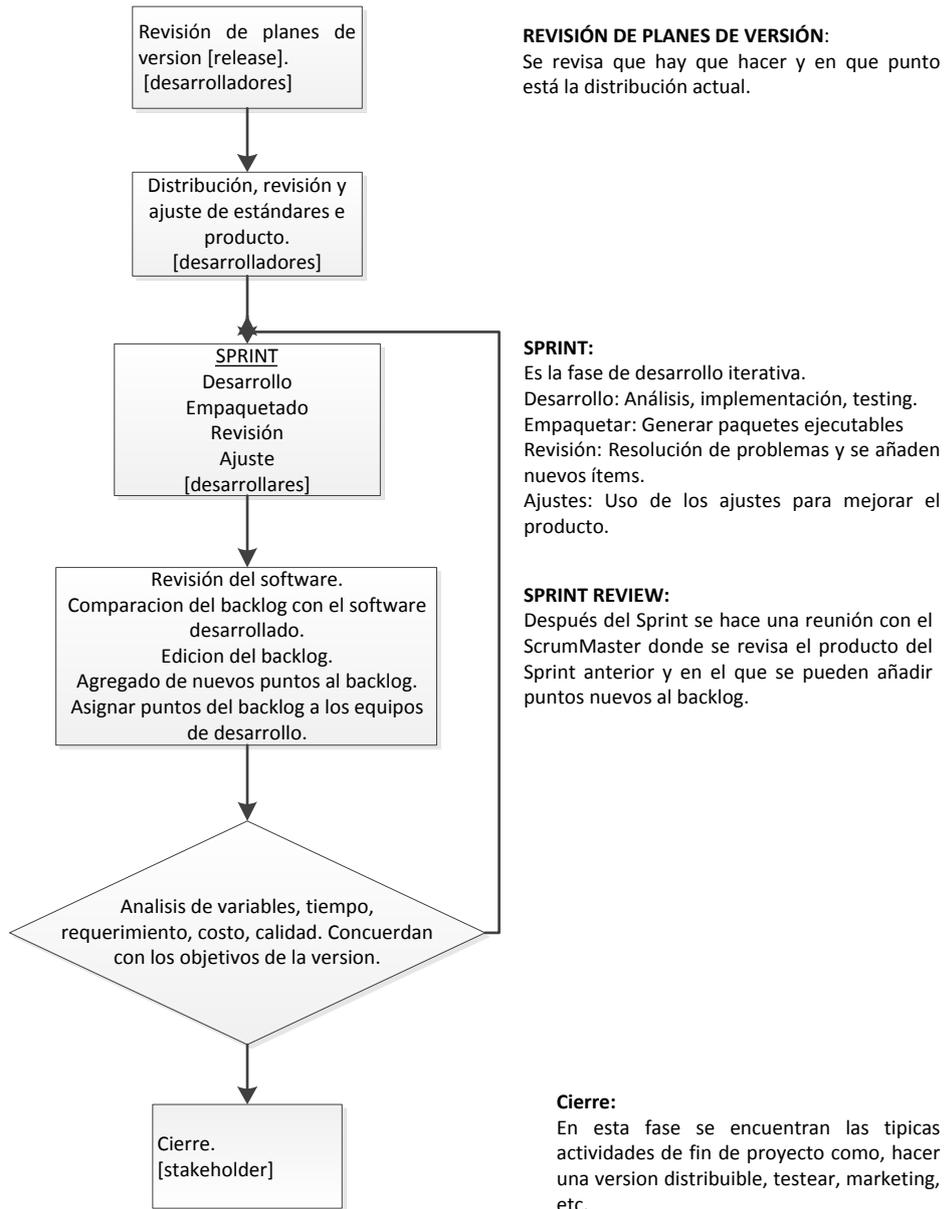


Figura V.43: Fases metodología SCRUM

Fuente: <http://osl2.uca.es/wikiCE/index.php/Scrum>

5.9 METODOLOGÍA SCRUM APLICADA AL DESARROLLO DEL SISTEMA ACADÉMICO DE “CONDUESPOCH”

A continuación se procede a especificar los requerimientos funcionales del sistema académico para CONDUESPOCH, también se detalla los Sprints con los que cuenta el sistema, dentro de cada uno se asigna las historias de usuario y las tareas a realizar en cada sprint.

REQUERIMIENTOS FUNCIONALES

Para identificar cada uno de los requerimientos se usará el formato que a continuación se detalla.

- REQ<<”-“>><<M>><<”Primera letra de cada palabra”>><<”Numero secuencial”>>

Basada en la especificación anterior, se presenta como quedaría el nombre para el módulo de inscripción

- REQ-MPI-01

Requerimientos del módulo de inscripción

Los requerimientos del módulo de inscripción son:

- **REQ-MI-01**-El sistema permitirá registrar los datos del estudiante
- **REQ-MI-02**-El sistema permitirá modificar los datos del estudiante inscrito.
- **REQ-MI-03**-El sistema permitirá buscar y listar los datos del estudiante.
- **REQ-MI-04**-El sistema permitirá validar todos los requisitos a cumplir del estudiante.
- **REQ-MI-05**-El sistema permitirá habilitar el periodo de inscripción mediante intervalos de tiempo establecidos.

Requerimientos del módulo matriculación

- **REQ-MM-06**-El sistema permitirá registrar una matrícula
- **REQ-MM-07**-El sistema permitirá modificar los datos de la matrícula.
- **REQ-MM-08**-El sistema permitirá buscar y listar los estudiantes matriculados según el curso en el periodo académico actual.

- **REQ-MM-09**-El sistema permitirá registrar matriculas dentro de un determinado intervalo de tiempo previamente establecido.

Requerimientos del módulo Notas

- **REQ-MN-10**-El sistema permitirá ingresar, modificar las notas parciales de un estudiante dentro de un periodo de tiempo previamente establecido.
- **REQ-MN-11**-El sistema permitirá ingresar, modificar los intervalos de tiempo en los cuales los docentes podrán ingresar las notas de los estudiantes.
- **REQ-MN-12**-El sistema permitirá registrar el porcentaje de asistencias de cada estudiante.
- **REQ-MN-13**-El sistema permitirá consultar y listar las notas de los estudiantes.
- **REQ-MN-14**-El sistema permitirá generar un reporte de notas parciales de los estudiantes agrupados por docente, paralelo en el periodo académico actual.
- **REQ-MN-15**-El sistema permitirá sacar promedio de notas parciales.
- **REQ-MN-16**-El sistema permitirá mostrar los estudiantes aprobados, reprobados en un determinado periodo académico.
- **REQ-MN-17**-El sistema permitirá generar un reporte de los estudiantes aprobados, reprobados en el periodo académico actual.
- **REQ-MN-18**-El sistema permitirá generar un reporte del porcentaje de las asistencias de los estudiantes en el periodo académico actual.
- **REQ-MN-19**-El sistema permitirá generar un reporte general de las notas parciales por materia, de los estudiantes en el periodo académico actual.
- **REQ-MN-20**- El sistema permitirá ingresar, modificar las notas de suspenso de un estudiante dentro de un periodo de tiempo previamente establecido.
- **REQ-MN-21**- El sistema permitirá ingresar, modificar las notas de grado de un estudiante dentro de un periodo de tiempo previamente establecido.

Requerimientos del módulo de administración general

- **REQ-MA-22**-El sistema permitirá ingresar y modificar secciones (matutina, vespertina).

- **REQ-MA-23**-El sistema permitirá ingresar y modificar docentes.
- **REQ-MA-24**-El sistema permitirá ingresar y modificar los requisitos necesarios para la matriculación de los estudiantes.
- **REQ-MA-25**-El sistema permitirá ingresar y modificar los subperiodos que contendrá el periodo académico actual.
- **REQ-MA-26**-El sistema permitirá ingresar y modificar el horario de clases.
- **REQ-MA-27**-El sistema permitirá ingresar y modificar las materias
- **REQ-MA-28**-El sistema permitirá registrar y modificar los intervalos de tiempo para el ingreso de las notas parciales.
- **REQ-MA-29**-El sistema permitirá registrar y modificar las materias de grado para el periodo académico actual.
- **REQ-MA-30**-El sistema permitirá que el docente visualice su horario para el periodo académico actual.
- **REQ-MA-31**-El sistema permitirá registrar y modificar a los usuarios del sistema.
- **REQ-MA-32**-El sistema permitirá ingresar y modificar los periodos académicos.
- **REQ-MA-33**-El sistema permitirá asignar materias a los docentes.
- **REQ-MA-34**-El sistema permitirá generar un reporte con el promedio general de las notas las materias y el promedio general de las notas de grado.

A continuación se procede a especificar los requerimientos del sistema académico para CONDUESPOCH, en base a la metodología scrum cada uno de ellos se detallan en los Sprints correspondientes con los que cuenta el sistema.

PRODUCT BACKLOG

La **Tabla V.XXXV** muestra el product backlog y sus historias de usuario que se definieron para el desarrollo de la aplicación, a cada historia de usuario se le asignó un 'valor' estimado en el rango de 1-10 correspondiente a la importancia que esta historia tiene para la empresa, también se asignó un 'valor' estimado correspondiente al esfuerzo que implica el desarrollo de la historia.

Tabla V.XXXV: Product Backlog

Fuente: Johnny Parra, Mario Andrade

ID	Historia de usuario	Valor	Puntos	Estado	Responsable	Iteración
H1	Como equipo de desarrollo se necesita analizar los requerimientos del cliente. Para limitar el alcance del sistema	10	14	Terminado	jparra, mandrade.	1
H2	Como equipo de desarrollo se necesita crear la base de datos	10	19	Terminado	Jparra, mandrade.	2
H3	Como secretaria deseo poder inscribir a un estudiante en el sistema	10	16	Terminado	Jparra, mandrade.	2
H4	Como secretaria deseo poder modificar los datos de un estudiante	7	17	Terminado	Jparra, mandrade.	2
H5	Como secretaria deseo poder visualizar los estudiantes previamente inscritos.	6	16	Terminado	Jparra, mandrade.	2
H6	Como administrador del sistema deseo poder crear usuarios	5	16	Terminado	Jparra, mandrade.	2
H7	Como secretaria deseo poder ver los requisitos que deben cumplir los estudiantes previo al proceso de matriculación.	5	17	Terminado	Jparra, mandrade.	3

H8	Como secretaria deseo poder matricular un estudiante en el sistema.	10	16	Terminado	Jparra, mandrade.	3
H9	Como secretaria deseo poder modificar los datos de una determinada matrícula.	5	16	Terminado	Jparra, mandrade.	3
H10	Como secretaria deseo poder visualizar la lista de estudiantes matriculados.	5	16	Terminado	Jparra, mandrade.	3
H11	Como director/a pedagógico deseo poder asignar materias a los docentes.	8	16	Terminado	Jparra, mandrade.	3
H12	Como director/a pedagógico deseo poder asignar la carga horaria a un docente.	8	17	Terminado	Jparra, mandrade.	3
H13	Como director/a pedagógico deseo poder visualizar la carga horaria de un docente.	6	16	Terminado	Jparra, mandrade.	3
H14	Como docente deseo poder ver mi horario de clases.	6	16	Terminado	Jparra, mandrade.	3
H15	Como inspector deseo poder registrar las asistencias de los estudiantes por materias y cursos.	10	19	Terminado	Jparra, mandrade.	4

H16	Como inspector deseo poder visualizar las asistencias de los estudiantes por materias y cursos.	7	16	Terminado	Jparra, mandrade.	4
H17	Como docente deseo poder registrar las notas parciales por curso y materia.	10	18	Terminado	Jparra, mandrade.	4
H18	Como docente deseo poder visualizar las notas parciales por curso y materia.	7	19	Terminado	Jparra, mandrade.	4
H19	Como docente deseo poder registrar las notas de suspenso por curso y materia.	8	19	Terminado	Jparra, mandrade.	4
H20	Como docente deseo poder visualizar las notas de suspenso por curso y materia.	6	16	Terminado	Jparra, mandrade.	4
H21	Como secretaria deseo poder registrar las notas de grado por curso y materia.	8	19	Terminado	Jparra, mandrade.	4
H22	Como secretaria deseo poder visualizar las notas de grado por curso y materia.	6	16	Terminado	Jparra, mandrade.	4
H23	Como equipo de desarrollo deseo poder realizar la parte de autenticación del sistema	10	20	Terminado	Jparra, mandrade.	5
H24	Como equipo de desarrollo deseo	8	16	Terminado	Jparra, mandrade.	5

	poder manejar perfiles de usuario.					
H25	Como usuario registrado en el sistema deseo poder modificar mis datos de acceso al mismo.	7	16	Terminado	Jparra, mandrade.	5
H26	Como administrador del sistema deseo poder ingresar un nuevo periodo académico.	10	16	Terminado	Jparra, mandrade.	5
H27	Como administrador del sistema deseo poder modificar el actual periodo académico.	6	16	Terminado	Jparra, mandrade.	5
H28	Como administrador del sistema deseo poder ingresar, modificar y listar los parciales a evaluar en un periodo académico.	8	19	Terminado	Jparra, mandrade.	5
H29	Como administrador del sistema deseo poder establecer el rango de fechas en que se podrán ingresar cada una de las evaluaciones parciales.	9	19	Terminado	Jparra, mandrade.	5
H30	Como administrador del sistema deseo poder establecer los días y las horas que serán utilizadas para la asignación de la	9	18	Terminado	Jparra, mandrade.	5

	carga horaria de docentes.					
H31	Como administrador del sistema deseo poder ingresar y modificar los subperiodos que contendrá el periodo actual.	9	16	Terminado	Jparra, mandrade.	5
H32	Como administrador del sistema deseo poder ingresar, modificar y listar las materias en el sistema.	9	18	Terminado	Jparra, mandrade.	5
H33	Como administrador del sistema deseo poder modificar de manera simultánea todas las materias previamente registradas en el sistema.	8	16	Terminado	Jparra, mandrade.	5
H34	Como administrador del sistema deseo poder ingresar y modificar las materias de grado que serán evaluadas en el periodo actual.	8	18	Terminado	Jparra, mandrade.	5
H35	Como administrador del sistema deseo poder asignar y modificar las materias a los distintos subperiodos previamente registrados.	8	16	Terminado	Jparra, mandrade.	5

H36	Como administrador del sistema deseo poder ingresar, modificar y listar: paralelos, niveles y cursos.	8	19	Terminado	Jparra, mandrade.	5
H37	Como administrador del sistema deseo poder ingresar, modificar y listar los requisitos que serán tomados en cuenta para el proceso de matriculación.	8	19	Terminado	Jparra, mandrade.	5
H38	Como administrador del sistema deseo poder modificar de manera simultánea todos los requisitos previamente registrados.	7	17	Terminado	Jparra, mandrade.	5
H39	Como administrador del sistema deseo poder ingresar, modificar y listar los roles en el sistema.	8	18	Terminado	Jparra, mandrade.	5
H40	Como docente deseo poder generar un reporte de los estudiantes que se quedaron en suspenso.	10	17	Terminado	Jparra, mandrade.	6
H41	Como docente deseo poder generar un reporte de cada una de las notas parciales.	8	17	Terminado	Jparra, mandrade.	6

H42	Como docente deseo poder generar un reporte general de calificaciones de las notas parciales.	10	18	Terminado	Jparra, mandrade.	6
H43	Como inspector deseo poder generar un reporte de las asistencias por curso y materia.	10	17	Terminado	Jparra, mandrade.	6
H44	Como secretaria deseo poder generar un reporte de que muestre el promedio general de materias, el promedio de exámenes de grado de cada estudiante.	10	18	Terminado	Jparra, mandrade.	6

SPRINT BACKLOG

La **Tabla V.XXXVI** muestra el sprint backlog y sus historias de usuario que se definieron para el desarrollo de la aplicación, a cada historia de usuario se le asignó un 'valor' estimado en el rango de 1-10 correspondiente a la importancia que esta historia tiene para la empresa.

Tabla V.XXXVI: Sprint Backlog

Fuente: Johnny Parra, Mario Andrade

Sprint	Historia de usuario	Valor	Puntos	Estado	Responsable
1	Como equipo de desarrollo. Necesitamos analizar los requerimientos del cliente. Para limitar el alcance del sistema	10	14	Terminado	Jparra, mandrade.
2	Como equipo de desarrollo se necesita crear la base de datos	10	19	Terminado	Jparra, mandrade.
	Como secretaria deseo poder inscribir a un estudiante en el sistema	10	16	Terminado	Jparra, mandrade.
	Como secretaria deseo poder modificar los datos de un estudiante	7	17	Terminado	Jparra, mandrade.
	Como secretaria deseo poder visualizar los estudiantes previamente inscritos.	6	16	Terminado	Jparra, mandrade.
	Como administrador del sistema deseo poder crear usuarios	5	16	Terminado	Jparra, mandrade.
3	Como secretaria deseo poder matricular un estudiante en el sistema.	10	16	Terminado	Jparra, mandrade.
	Como director/a pedagógico deseo poder asignar materias a los docentes.	8	16	Terminado	Jparra, mandrade.
	Como director/a pedagógico deseo poder asignar la carga horaria a un docente.	8	17	Terminado	Jparra, mandrade.
	Como secretaria deseo poder ver los requisitos que	5	17	Terminado	Jparra, mandrade.

	deben cumplir los estudiantes previo al proceso de matriculación.				
	Como secretaria deseo poder modificar los datos de una determinada matrícula.	5	16	Terminado	Jparra, mandrade.
	Como secretaria deseo poder visualizar la lista de estudiantes matriculados.	5	16	Terminado	Jparra, mandrade.
	Como director/a pedagógico deseo poder visualizar la carga horaria de un docente.	6	16	Terminado	Jparra, mandrade.
	Como docente deseo poder ver mi horario de clases.	6	16	Terminado	Jparra, mandrade.
4	Como inspector deseo poder registrar las asistencias de los estudiantes por materias y cursos.	10	19	Terminado	Jparra, mandrade.
	Como docente deseo poder registrar las notas parciales por curso y materia.	10	18	Terminado	Jparra, mandrade.
	Como docente deseo poder registrar las notas de suspenso por curso y materia.	8	19	Terminado	Jparra, mandrade.
	Como secretaria deseo poder registrar las notas de grado por curso y materia.	8	19	Terminado	Jparra, mandrade.
	Como inspector deseo poder visualizar las asistencias de los estudiantes por materias y cursos.	7	16	Terminado	Jparra, mandrade.
	Como docente deseo poder visualizar las notas parciales por curso y materia.	7	19	Terminado	Jparra, mandrade.
	Como docente deseo poder visualizar las notas de	6	16	Terminado	Jparra, mandrade.

	suspense por curso y materia.				
	Como secretaria deseo poder visualizar las notas de grado por curso y materia.	6	16	Terminado	Jparra, mandrade.
5	Como equipo de desarrollo deseo poder realizar la parte de autenticación del sistema	10	20	Terminado	Jparra, mandrade.
	Como administrador del sistema deseo poder ingresar un nuevo periodo académico.	10	16	Terminado	Jparra, mandrade.
	Como administrador del sistema deseo poder establecer el rango de fechas en que se podrán ingresar cada una de las evaluaciones parciales.	9	19	Terminado	Jparra, mandrade.
	Como administrador del sistema deseo poder establecer los días y las horas que serán utilizadas para la asignación de la carga horaria de docentes.	9	18	Terminado	Jparra, mandrade.
	Como administrador del sistema deseo poder ingresar y modificar los subperiodos que contendrá el periodo actual.	9	16	Terminado	Jparra, mandrade.
	Como administrador del sistema deseo poder ingresar, modificar y listar las materias en el sistema.	9	18	Terminado	Jparra, mandrade.
	Como equipo de desarrollo deseo poder manejar perfiles de usuario.	8	16	Terminado	Jparra, mandrade.
	Como administrador del sistema deseo poder	8	19	Terminado	Jparra, mandrade.

ingresar, modificar y listar los parciales a evaluar en un periodo académico.				
Como administrador del sistema deseo poder modificar de manera simultánea todas las materias previamente registradas en el sistema.	8	16	Terminado	Jparra, mandrade.
Como administrador del sistema deseo poder ingresar y modificar las materias de grado que serán evaluadas en el periodo actual.	8	18	Terminado	Jparra, mandrade.
Como administrador del sistema deseo poder asignar y modificar las materias a los distintos subperiodos previamente registrados.	8	16	Terminado	Jparra, mandrade.
Como administrador del sistema deseo poder ingresar, modificar y listar: paralelos, niveles y cursos.	8	19	Terminado	Jparra, mandrade.
Como administrador del sistema deseo poder ingresar, modificar y listar los requisitos que serán tomados en cuenta para el proceso de matriculación.	8	19	Terminado	Jparra, mandrade.
Como administrador del sistema deseo poder ingresar, modificar y listar los roles en el sistema.	8	18	Terminado	Jparra, mandrade.
Como administrador del sistema deseo poder modificar de manera simultánea todos los	7	17	Terminado	Jparra, mandrade.

	requisitos previamente registrados.				
	Como administrador del sistema deseo poder modificar el actual periodo académico.	6	16	Terminado	Jparra, mandrade.
6	Como docente deseo poder generar un reporte de los estudiantes que se quedaron en suspenso.	10	17	Terminado	Jparra, mandrade.
	Como docente deseo poder generar un reporte general de calificaciones de las notas parciales.	10	18	Terminado	Jparra, mandrade.
	Como inspector deseo poder generar un reporte de las asistencias por curso y materia.	10	17	Terminado	Jparra, mandrade.
	Como secretaria deseo poder generar un reporte de que muestre el promedio general de materias, el promedio de exámenes de grado de cada estudiante.	10	18	Terminado	Jparra, mandrade.
	Como docente deseo poder generar un reporte de cada una de las notas parciales.	8	17	Terminado	Jparra, mandrade.

A continuación se detalla el primer sprint del sistema, dentro del cual se asignó las historias de usuario y a ellas las tareas que se realizaron en el mismo.

• **SPRINT 1**

La **tabla V.XXXVII** detalla el primer sprint 1 del sistema, en el cual se detalla las fechas de inicio y fin del sprint, una breve descripción sobre las tareas a desarrollarse en este sprint, los responsables y el tiempo asignado para su desarrollo.

Tabla V.XXXVII: Sprint 1

Fuente: Johnny Parra, Mario Andrade

SPRINT 1	
Fecha Inicio:	4/6/2014
Fecha Fin:	12/06/2014
Descripción:	Estudio de requerimientos, Definición y delimitación del alcance del proyecto
Esfuerzo estimado:	7 días
Responsable:	Johnny Parra, Mario Andrade

La **tabla V.XXXVIII** detalla la primera historia de usuario correspondiente al sprint 1 del sistema, en la cual se detalla el número asignado a la historia, su nombre, el tipo de usuario, la prioridad en el negocio, su riesgo de desarrollo y una breve descripción sobre la misma.

Tabla V.XXXVIII: Historia de usuario 1 del sprint 1

Fuente: Johnny Parra, Mario Andrade

HISTORIA DE USUARIO	
Número: H1	Nombre de la historia: Se analizará los requerimientos del cliente para delimitar el alcance del sistema.
Usuario: Desarrollador	Iteración Asignada: 1
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 14
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como equipo de desarrollo. Necesitamos analizar los requerimientos del cliente. Para limitar el alcance del sistema.	
Observaciones: <ul style="list-style-type: none"> • Es necesario tener reuniones con el personal que va a hacer uso del sistema para tener una mejor visión de lo que se requiere. 	

La **tabla V.XXXIX** detalla la primera tarea correspondiente a la historia de usuario 1, en la cual se detalla su número, el número de la historia a la cual pertenece, el tipo de tarea, el tiempo asignado para su desarrollo, el responsable de la tarea y una breve descripción sobre cómo se llevó a cabo la misma.

Tabla V.XXXIX: Tarea 1 de la historia de usuario 1

Fuente: Johnny Parra, Mario Andrade

TAREA DE INGENIERÍA	
Historia de Usuario: H1	
Número Tarea: 1	Nombre de la Tarea: Reuniones con el personal de la institución.
Tipo de Tarea: Análisis de requerimientos	Puntos de Estimados: 8
Fecha de Inicio: 04/06/2014	Fecha de Fin: 09/06/2014
Responsables: Johnny Parra, Mario Andrade	
Descripción: <ul style="list-style-type: none">• Se tendrá algunas reuniones con el personal que va a hacer uso del sistema, a fin de extraer de modo más preciso los requerimientos.	

Las demás tareas del sprint 1, junto con los demás Sprints que se utilizaron en el desarrollo del sistema se encuentran en el **Anexo VI**.

La **figura V.44** ilustra la frecuencia en que se cumplían las tareas de cada uno de los Sprints que contiene el desarrollo del sistema, la misma se obtuvo del repositorio GitHub en el cual está almacenado el sistema.

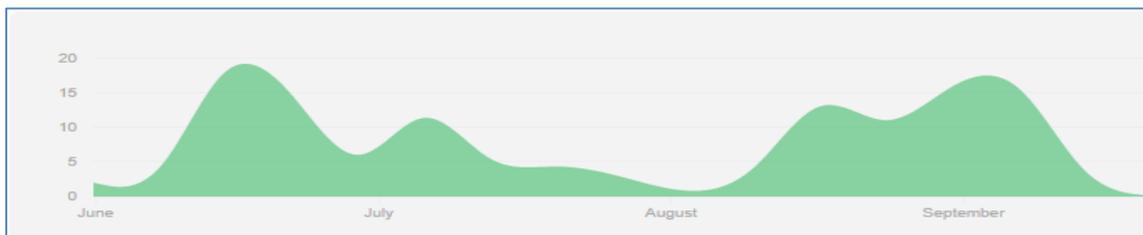


Figura V.44: Fases metodología SCRUM

Fuente: <https://github.com/mariofernip/conduespoch/graphs/contributors?from=2014-06-01&to=2014-09-29&type=c>

SISTEMA ACADÉMICO CONDUESPOCH BURN UP

La **figura V.45** muestra las 6 iteraciones definidas para el desarrollo de la aplicación, cada iteración tiene una fecha de inicio y de fin, los puntos representan el avance en cada iteración.

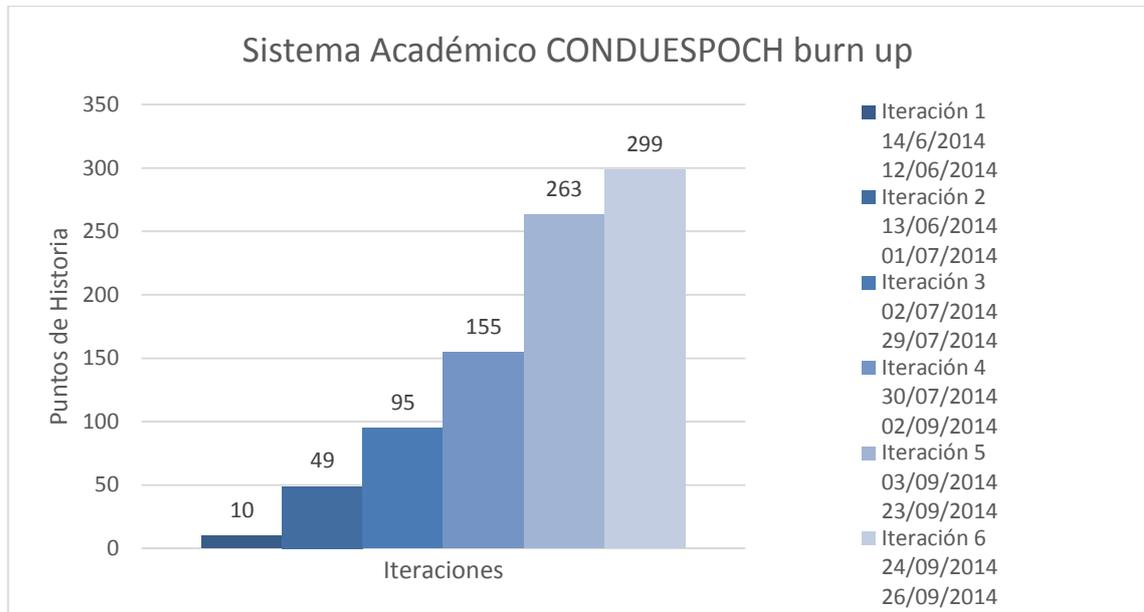


Figura V.45: Sistema Académico CONDUESPOCH burn up

Fuente: Johnny Parra, Mario Andrade

CODIFICACIÓN DEL SISTEMA

Para desarrollar el sistema se empleó Symfony2 versión 2.3 el popular framework para desarrollar aplicaciones PHP. Entre sus principales características cabe destacar su facilidad de aprendizaje, es libre, además es uno de los frameworks PHP con mejor rendimiento.

Para el desarrollo se utilizó la arquitectura MVC, la cual tiene como finalidad separar los datos de la lógica de negocio, se utilizó tres componentes como: el modelo, vista y controlador. Cada capa fue creada en directorios, los mismos que contenían archivos .php, .twig.

DISEÑO DE INTERFACES DE USUARIO PARA EL SISTEMA ACADEMICO CONDUESPOCH

En esta sección se describe de manera general el diseño y los elementos que forman parte del sistema académico CONDUESPOCH, también se detalla brevemente el diseño de los reportes que presenta el sistema.

Para la realización de las vistas se utilizó las siguientes características:

Banner.- Contiene una imagen en la parte superior, dentro de la misma se encuentra el logo de la institución en la parte izquierda, en el centro se coloca tanto el nombre del periodo académico actual y en la parte derecha un enlace para modificar los datos del perfil de usuario.

Menú.- Permiten agrupar opciones de navegación para la aplican web. Se realizó mediante listas no ordenadas UL y etiquetas LI.

Campos requeridos.- son cuadros de textos necesarios para el ingreso o actualización, estos se resaltaron con un fondo de color gris.

Listas desplegables.- para el registro o actualización de datos como la hora, día, año entre otros, se utilizaron listas desplegable con la etiqueta SELECT.

Botones de Comando.- Permiten finalizar tareas tales como registrar, actualizar. Se lo realizo mediante un input type submit.

Sección aside.- En la parte derecha de la aplicación se colocó una región aside en donde se mostró el enlace correspondiente al cierre de sesión, también se muestran mensajes de respuesta del sistema y en caso del perfil de secretaria se mostró información relacionada al número de estudiante matriculados, inscritos.

Footer.- Se colocó en la parte inferior y centrado, información relacionada a la empresa como la dirección, teléfono.

Las características antes descritas características están ilustradas en todas las vistas del sistema.

La **Figura V.46** ilustra un ejemplo de una interfaz de sistema.

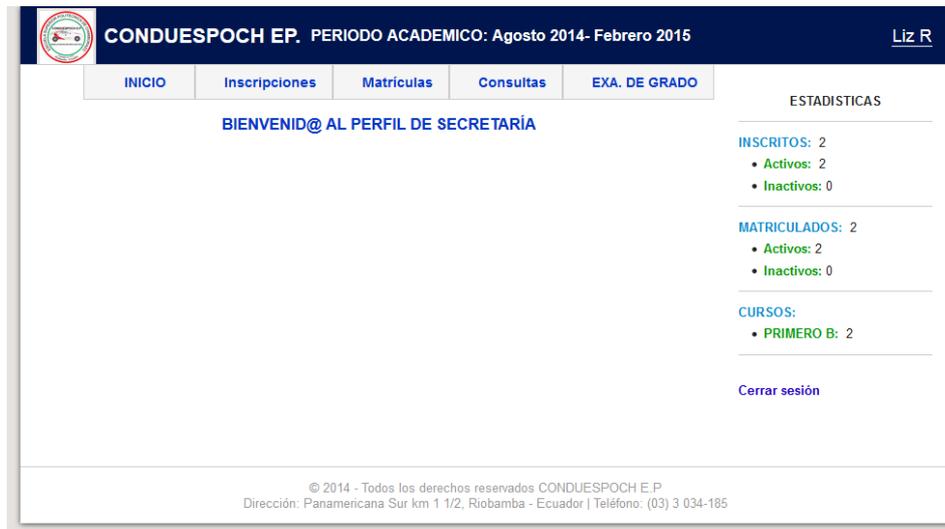


Figura V.46: Vista del sistema: Perfil Secretaria.

Fuente: Johnny Parra, Mario Andrade

Pantallas de Visualización de Reportes

Para el diseño de los reportes del sistema se tomó como referencia un modelo proporcionado por la empresa, a continuación se menciona las características generales de los reportes:

Encabezado.- Posee el logo de la empresa en la parte derecha, centrado se encuentra el nombre de la empresa, la ciudad y provincia. La fuente que se utilizó para estos textos fue "Lucida console-14", a continuación se encuentra el nombre del reporte, la fuente que se utilizó para este texto fue "Lucida console-15".

Cuerpo.- Consta de 2 partes, en la primera se encuentra una tabla con los datos correspondientes al nombre de la asignatura, nombre del docente, el curso y paralelo, la sección, el periodo actual. La fuente que se utilizó para estos textos es "Lucida console-14". En la segunda parte se encuentra una tabla con los datos del estudiante como nombre y apellido, las calificaciones y de ser el caso la suma y promedios. La fuente que se utilizó para estos textos es "Lucida console-13".

Los espacios para las firmas de acuerdo al número de estas, se las coloca a la izquierda o derecha inferior de cada documento. La fuente que se utilizó para estos textos es “Lucida console-13”.

En el pie de página se colocó la dirección de la empresa, el nombre de la institución centrado, en la parte izquierda se encuentra el número de página. La fuente que se utilizó para estos textos es “Lucida console-10”.

En la **Figura V.47** se puede observar un ejemplo de un reporte de acta de calificaciones.

ESCUELA DE CONDUCCIÓN PROFESIONAL
CONDUESPOCH E.P
CHIMBORAZO - RIOBAMBA



ACTA DE CALIFICACIONES

ASIGNATURA:	Educación Vial
DOCENTE:	Yony Par
CURSO:	PRIMERO
PARALELO:	B
SECCIÓN:	DIURNA
PERIODO:	AGOSTO 2014- FEBRERO 2015
CALIFICACIONES DE:	PARCIAL 1: noviembre noviembre

Nº	NOMINA	Trabajo en equipo	Estudio de casos	Prueba práctica	Prueba teórica	SUMA	PROMEDIO
1	SALLE JOSE	20.00	20.00	20.00	20.00	80.00	20.00
2	TOBBA JULIO	20.00	20.00	18.00	20.00	78.00	19.50

(R) DIRECTOR(A) PEGAGÓGICO

(R) DOCENTE

Dirección: Panamericana Sur km 1.52, Riobamba - Ecuador
©Sistema conduespoch

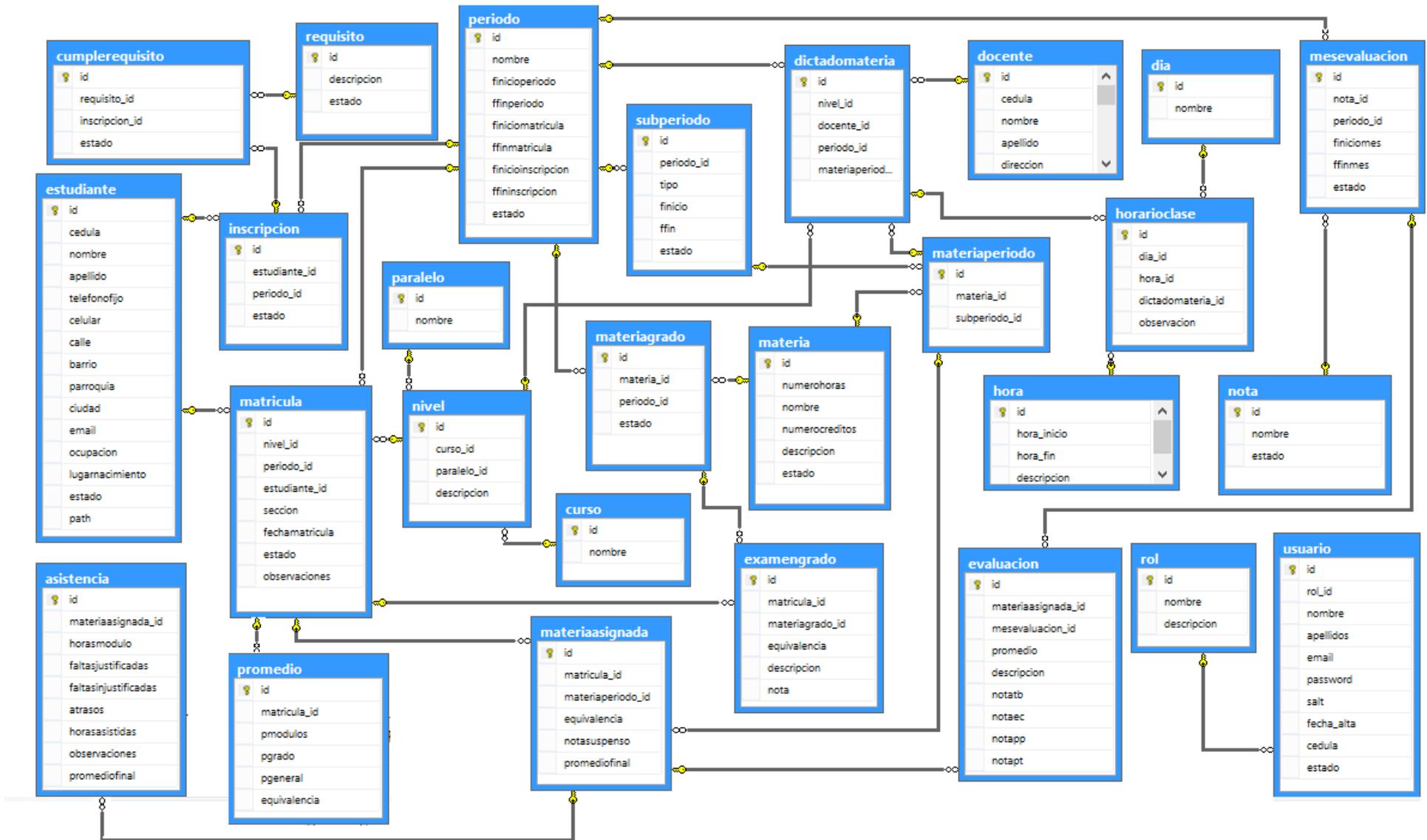
1/1

Figura V.47: Reporte de acta de calificaciones.

Fuente: Johnny Parra, Mario Andrade

La especificación del resto de reportes y demás interfaces se encuentra en el Manual de Usuario.

ESQUEMA DE BASE DE DATOS DEL SISTEMA



CONCLUSIONES

- En base al estudio comparativo se deduce que el SGBD SQL Server Express Edition 2012 tiene un mejor rendimiento en comparación al SGBD PostgreSQL 9.2.4 en un 48,72%.
- El SGBD SQL Server Express Edition 2012 en la prueba relacionada al número de órdenes por minuto se notó una estabilidad hasta con 900 usuarios, pasado este número se evidenció errores en el desarrollo de la prueba relacionado a la concurrencia de usuarios.
- De acuerdo a la documentación del SGBD SQL Server Express Edition 2012 la memoria máxima que puede usar el SGBD es de 1GB, sin embargo las pruebas realizadas registraron un uso superior a las 3GB de memoria RAM.
- El SGBD PostgreSQL 9.2.4 fue más veloz en 47,3% en comparación con el SGBD SQL Server Express Edition 2012 en las operaciones de consulta de registros.

RECOMENDACIONES

- Para entornos empresariales en donde se manejan gran cantidad de transacciones durante largos periodos de tiempo que demandan un gran uso de recursos como memoria RAM y capacidad de procesamiento que superen la limitaciones del SGBD SQL Server Express Edition 2012 se recomienda utilizar PostgreSQL 9.2.4 ya que este SGBD no tiene tal limitación.
- Para entornos en donde la base de datos supere los 10GB no es recomendable usar el SGBD SQL Server Express Edition 2012.
- Si la base de datos demanda una gran escalabilidad es recomendable utilizar el SGBD PostgreSQL 9.2.4 pues tiene mejores características que el SGBD SQL Server Express Edition 2012 en cuanto a escalabilidad y capacidad de almacenamiento.
- Realizar el mismo estudio en equipos con mejores prestaciones y utilizando algún SGBD SQL Server en sus versiones de pago, por ejemplo: SQL Server Enterprise Edition.

RESUMEN

Análisis comparativo entre PostgreSQL y SQL Server, con el propósito de medir el rendimiento en la implementación de aplicaciones web, aplicado al sistema Académico de CONDUESPOCH.

Para realizar el análisis comparativo se aplicó el método inductivo, los parámetros de medición que se definieron fueron el consumo de recursos, tiempo de respuesta, número de órdenes por minuto y tiempo del aseguramiento de la integridad de datos, para poder obtener los resultados se aplicó la técnica de comparación utilizando estadística descriptiva.

Las herramientas que se utilizaron para realizar el análisis fueron HammerDB, Manage Engine y se utilizó un prototipo de aplicación web. Los resultados porcentuales de los parámetros con respecto al consumo de recursos del SGBD PostgreSQL fue 64,57% y SQL Server obtuvo 80.09%, en el tiempo de respuesta del SGBD PostgreSQL fue 77.84% y SQL Server logró 72,20%, en relación al número de órdenes por minuto del SGBD PostgreSQL fue 12% y SQL Server alcanzó 100%, en el tiempo del aseguramiento de la integridad de datos del SGBD PostgreSQL fue 3,10% y SQL Server tuvo 100%. Finalmente en la tabulación de resultados el SGBD SQL Server obtuvo 88% equivalente a excelente y el SGBD PostgreSQL logró 39,34% equivalente a Regular.

Se concluye que el SGBD SQL Server tiene un mejor rendimiento en el desarrollo de aplicaciones web.

Se recomienda utilizar el SGBD SQL Server en las PYMES debido a que el volumen de información que se maneja en estas empresas no es muy grande.

SUMMARY

Comparative analysis between PostgreSQL and SQL Server, with the aim to gauge the performance in the implementing of web application, executed to the Academic System of CONDUESPOCH.

To make the comparative analysis was applied the inductive method, the measurement parameters defined were resource consumption, answer time, order number per minute and integrity assurance time of data, in order to be able to obtain the results were applied the comparison technique by using Descriptive Statistics .

The tools used to make the analysis were HammerDB, Manage Engine and it was applied a prototype of the web application. The porcentual results of the parameter in relation to the resource consumption of DBMS (Data Base Management System) PostgreSQL was 64,57% and SQL Server got a 80,09%, in the answer time of the DBMS PostgreSQL was 77,84% and SQL Server obtained a 72,20% relative to the order number per minute of the DBMS PostgreSQL was 12% and SQL Server reached to 100% in the integrity assurance time of data of the DBMS PostgreSQL was 3,10% and SQL Server had 100%. Finally, in the results tabulation the DBMS SQL Server got 88% equivalent to excellent and the DBMS PostgreSQL had 39,34% equivalent to regular.

It is concluded that the DBMS SQL Server has a better performance in the development of web applications.

It is recommended to use DBMS SQL Server in the PYMES(Small and Medium Enterprises) due to the data volume that is administered in these enterprises is not too big.

GLOSARIO

Administrador	Persona responsable del manejo del sistema.
Aplicación web	Es un conjunto de páginas que interactúan entre sí haciendo uso de los recursos contenidos en un servidor web, incluidas base de datos.
Atributo:	Representa una propiedad dentro de una entidad o tabla correspondiente a una base de datos.
Base de datos:	Es un conjunto de datos organizados y estrechamente relacionados entre sí, los mismos que son consumidos por sistemas informáticos dentro de empresas o un negocio en particular.
Clave primaria:	Es una clave única entre todos los atributos de una tabla, para especificar los datos relacionados con otras tablas.
Clave foránea:	No son únicas en una tabla y hacen referencia a una clave en otra tabla
Cliente:	Persona que hace uso del sistema con determinados privilegios.
Concurrencia:	Propiedad de los sistemas informáticos que permiten la ejecución de múltiples procesos al mismo tiempo e interactúan entre sí.
Entidad:	Cualquier objeto real o abstracto del mundo del cual se desea almacenar información en la base de datos.
Framework:	Es un marco de trabajo definido, donde dentro de su estructura otro proyecto de software puede ser desarrollado

Hardware:	Hace referencia a todas las partes físicas que conforman una computadora
Integridad de datos:	Hace referencia a los métodos de los datos que se almacenan y se utilizan en las estructuras de datos de manera correcta.
Interfaz:	Las ventanas con las que interactúa el usuario del sistema.
Javascript:	Lenguaje basado en objetos muy liviano comúnmente utilizado para acceder a los objetos en aplicaciones web, permitiendo así el desarrollo de aplicaciones de interfaces enriquecidas y dinámicas.
Modelo relacional:	Permite la representación de la información del mundo real de manera muy fácil e intuitiva haciendo uso de relaciones.
Registro:	Conjunto de información relacionada a un determinado objeto o persona
Servidor:	Equipo informático que provee un conjunto de servicios a otras computadoras (clientes).
Sistema informático:	Permite almacenar, procesar y ordenar información haciendo uso de partes del hardware y software que interactúan entre sí.
Software:	Parte intangible de una computadora que está compuesto por una serie de instrucciones y datos, que hace uso de los recursos de una computadora para realizar una serie de tareas específicas
Transacción:	Es una unidad única de trabajo que contiene varias operaciones en un único bloque

ANEXOS

ANEXO I

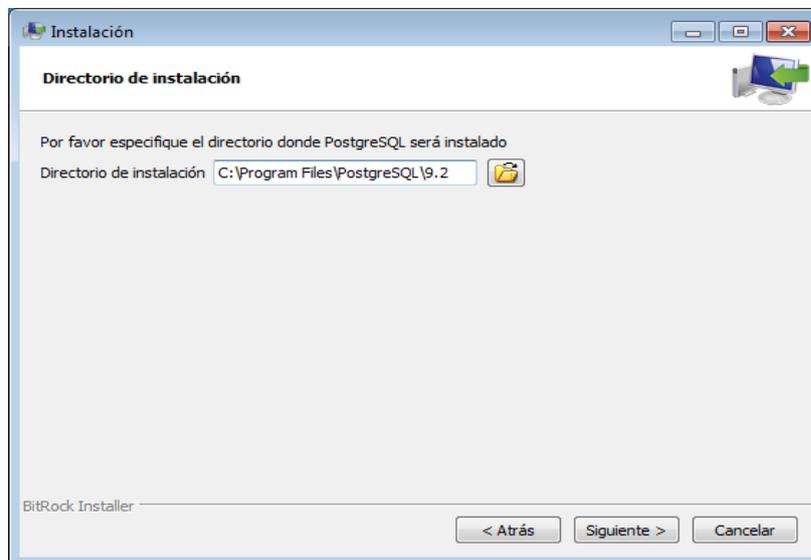
GUÍA DE INSTALACIÓN DE LOS SGBD POSTGRESQL Y SQL SERVER

GUÍA DE INSTALACIÓN DE POSTGRESQL

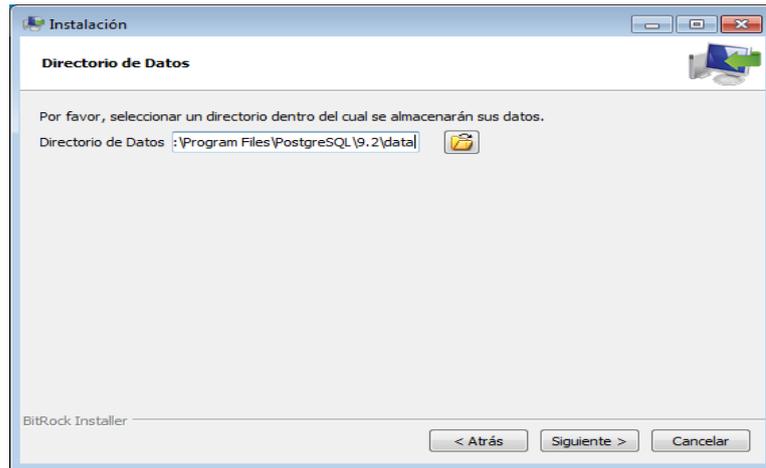
La primera pantalla que se nos muestra es la bienvenida al instalador de PostgreSQL. A partir de allí se tendrá que pulsar "Siguiente" cada vez que se requiera avanzar al siguiente paso.



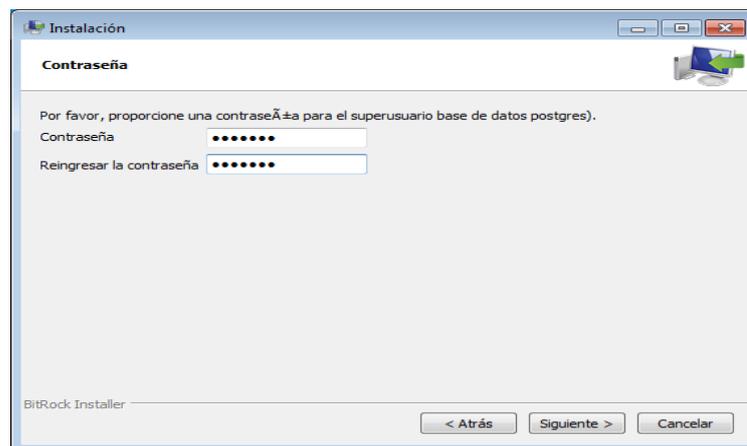
En el siguiente paso se tendrá que definir el directorio donde se va a instalar. En este caso, se utilizará el valor por defecto que el programa sugiere



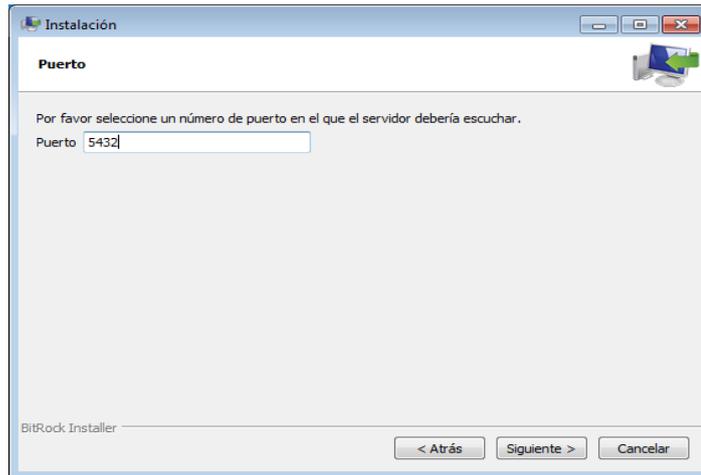
Una vez que se haya definido el directorio que será utilizado para instalar el programa, pasar al siguiente paso. En este paso de igual forma hay que definir el directorio de datos en donde se crearán las bases de datos. De nuevo, en este caso se utilizará el valor por defecto que el programa sugiere.



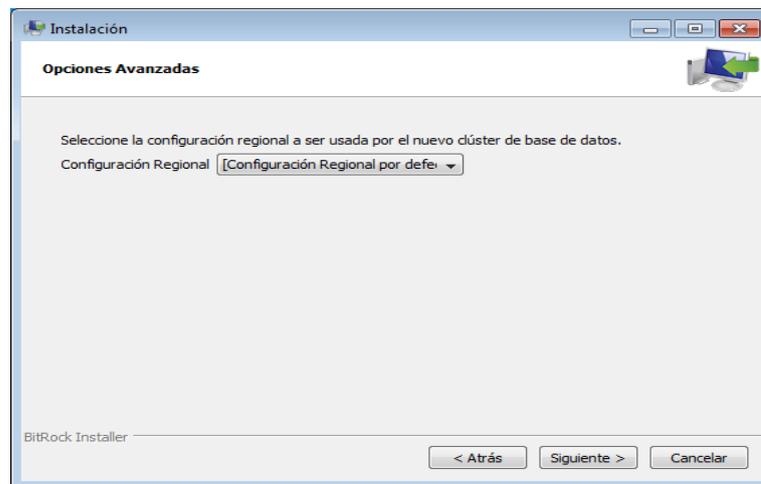
Pasar al siguiente paso. En el cual se tendrá que definir una clave de acceso para el usuario administrador de la base de datos PostgreSQL. Más adelante se podrá cambiar de ser necesario la clave del usuario administrador (postgres) de la base de datos. Esta clave es totalmente independiente de la clave de la cuenta de servicio 'postgres' en el sistema operativo.



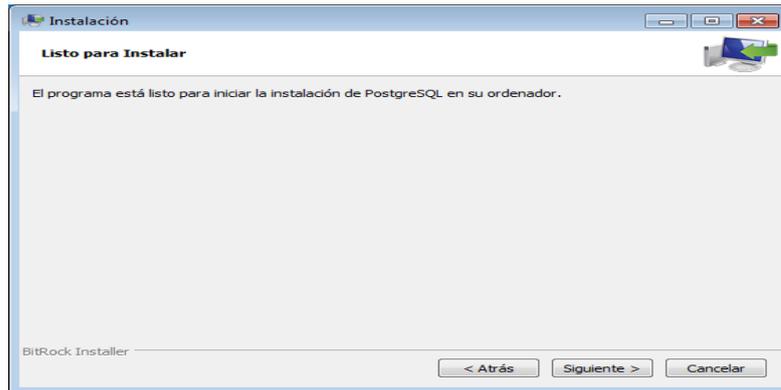
En esta instancia se procederá a especificar el puerto que PostgreSQL utilizará para escuchar las futuras conexiones. En este caso se dejará el valor por defecto, 5432. Pero vale la pena recalcar que para un ambiente de producción es una buena práctica cambiar el número de puerto designado por defecto al SGBD.



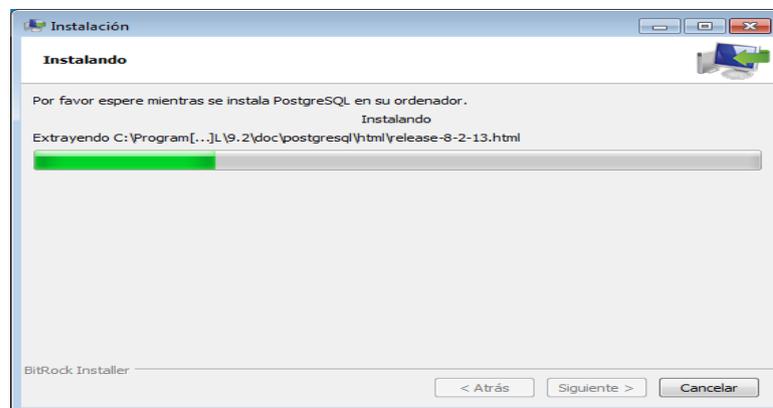
Una vez que hemos culminado con los pasos básicos, el programa entra en la sección de opciones avanzadas. En este paso hay que decidir que 'locale' se desea utilizar y si se quiere instalar PL/pgSQL en la base de datos template1. Se recomienda seguir utilizando la configuración por defecto que está ofreciendo el asistente de instalación.



Una vez que se ha culminado con todos los pasos de configuración, el programa informa que está listo para empezar a instalar e inicializar la presente instalación.



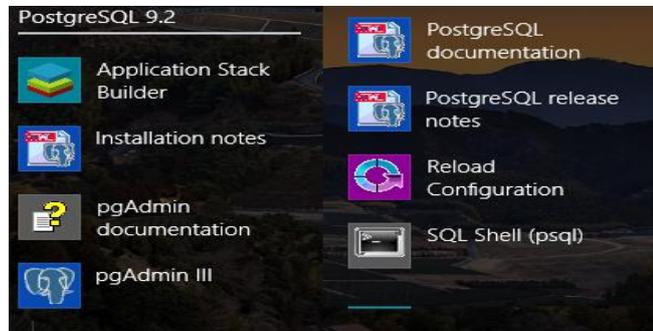
Pulsar por última vez "Siguiente" y esperar a que el programa termine de instalar todo.



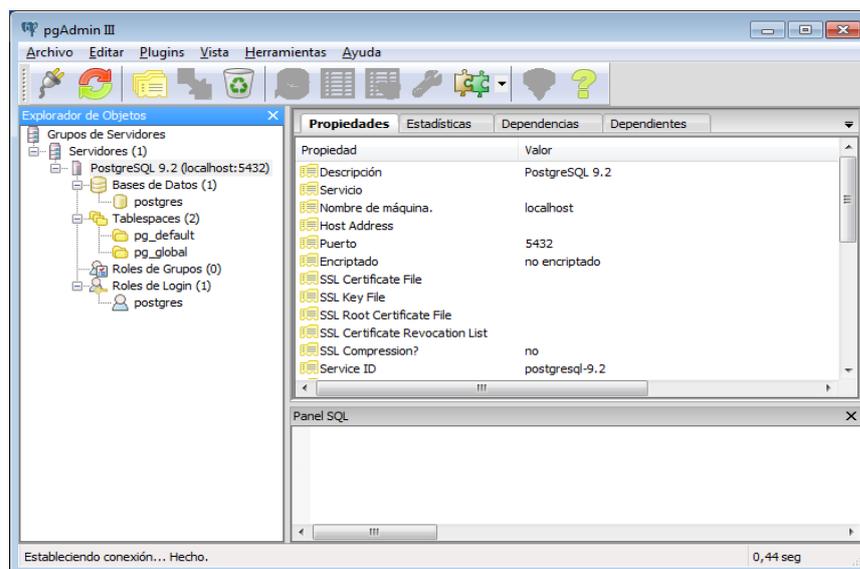
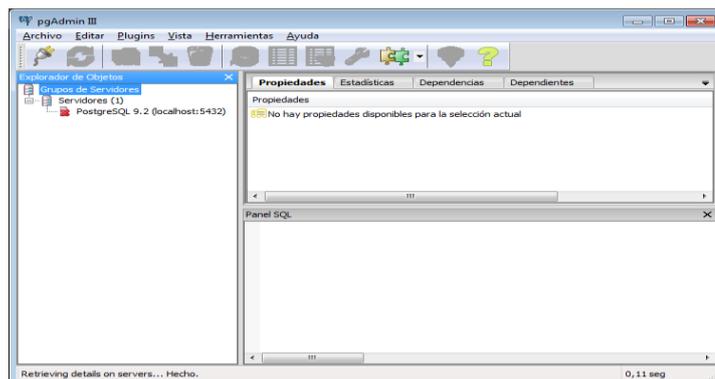
Una vez terminada la instalación, se podrá salir del instalador pulsando "Terminar". En este último paso el instalador nos presenta la opción de arrancar automáticamente un programa llamado "Stack Builder". Este programa puede ser usado para instalar diversos programas adicionales, tanto programas libres como algunos distribuidos por EnterpriseDB. Como aquello no es necesario para este efecto, deseleccionamos esa opción y pulsar en "Terminar".



Si todo el proceso de instalación ha salido bien se dispondrá de un nuevo menú en el sistema. Desde este menú se podrá acceder a la documentación de PostgreSQL, a diversos programas clientes, arrancar y apagar la base de datos, etc.



A continuación se presenta algunas pantallas del cliente de PostgreSQL **pgAdminIII** para el sistema operativo de Windows.



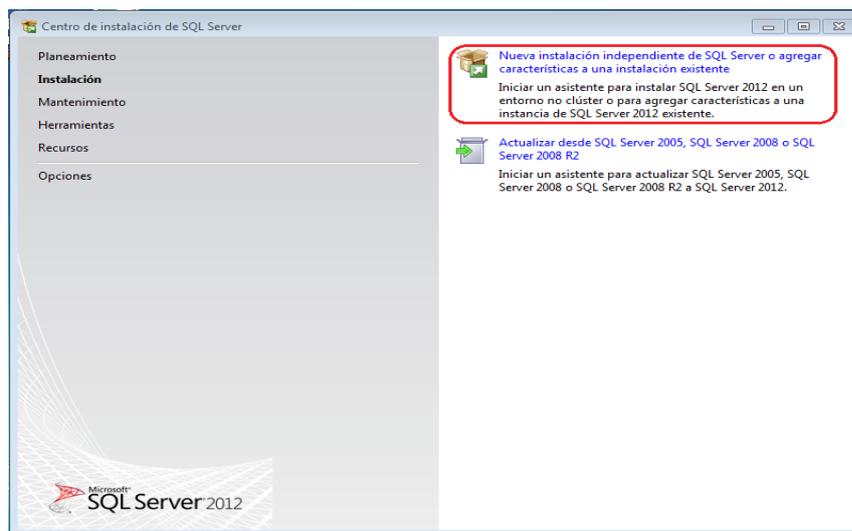
GUÍA DE INSTALACIÓN DE SQL SERVER 2012 EXPRESS

A continuación se presenta una guía sobre la instalación de SQL Server 2012 EXPRESS.

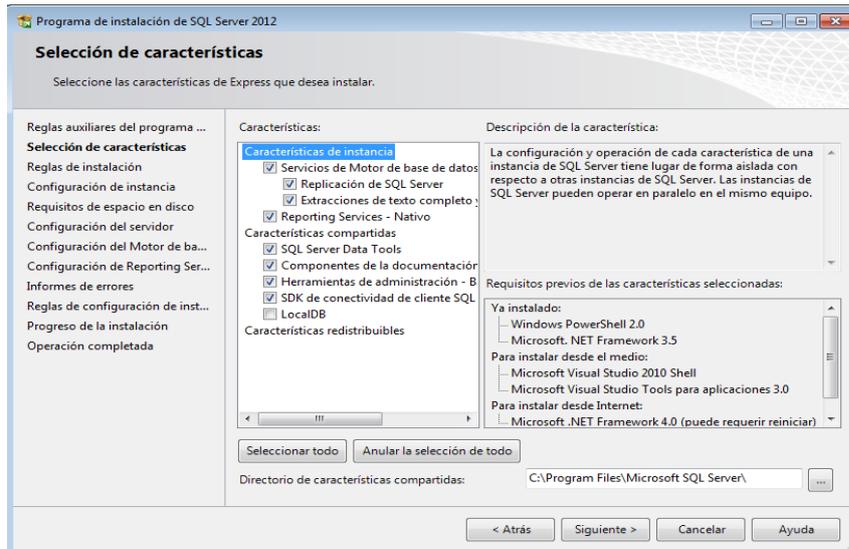
Dirigirse a la página del producto y descargar el instalador, entonces elegir entre cinco tipos distintos:

- SQL Server Express con Herramientas (con LocalDb, incluye el motor de base de datos y SQL Server Management Studio Express)
- SQL Server Management Studio (solo herramientas)
- SQL Server Express LocalDb (instalador MSI)
- SQL Server Express con Servicios Avanzados
- SQL Server Express (solo motor de base de datos).

El siguiente paso es lanzar el instalador para llegar al Centro de instalación de SQL Server.

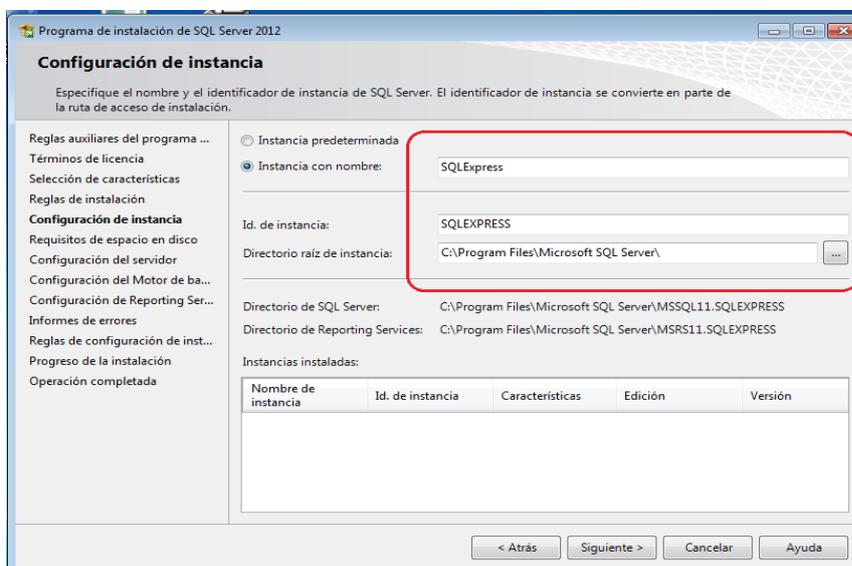


Dar clic en siguiente para buscar actualizaciones. A continuación copiar los archivos del programa de instalación, se comprueban las reglas auxiliares y elegir las características a instalar. Pulsar siguiente y empezar a copiar los archivos.

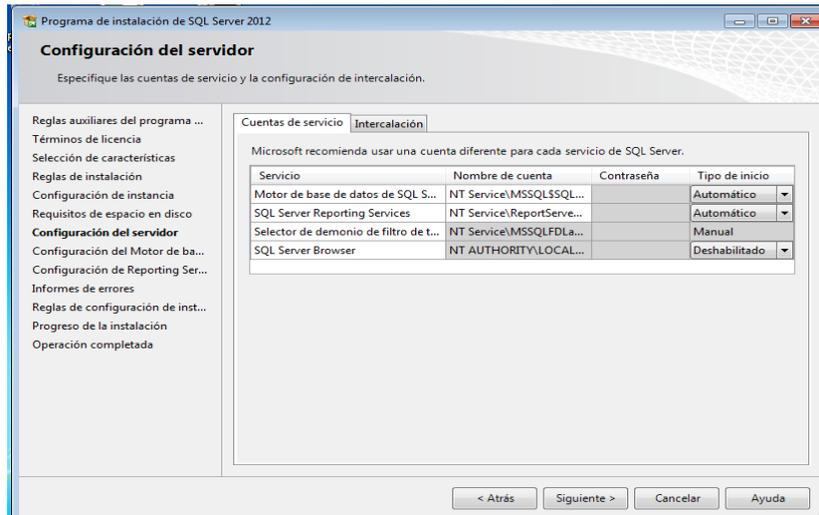


Como requisitos previos, se necesita .NET Framework 3.5 Service Pack 1 y poder acceder al centro de descargas para poder actualizar con la versión 4.0.

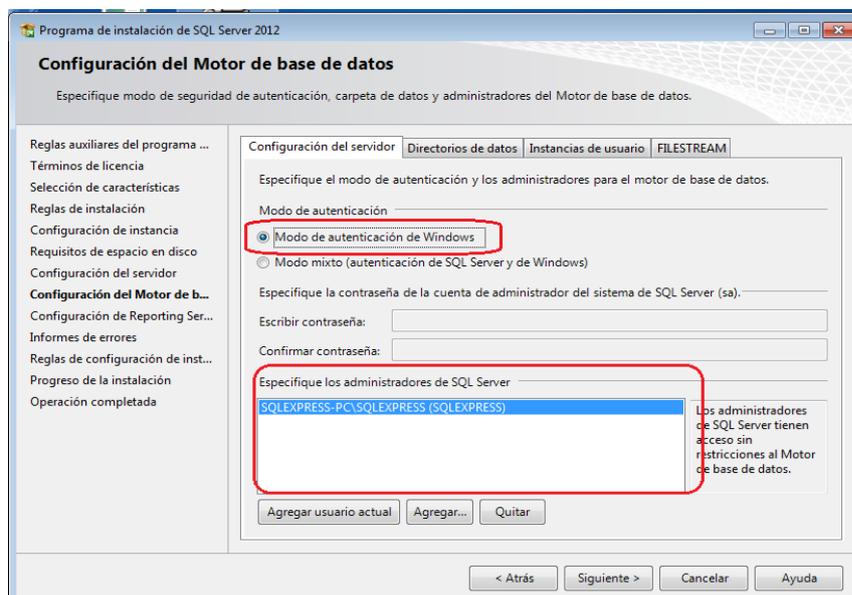
El paso siguiente a la elección de las características es la configuración de la instancia. Especificar el nombre y directorio raíz.



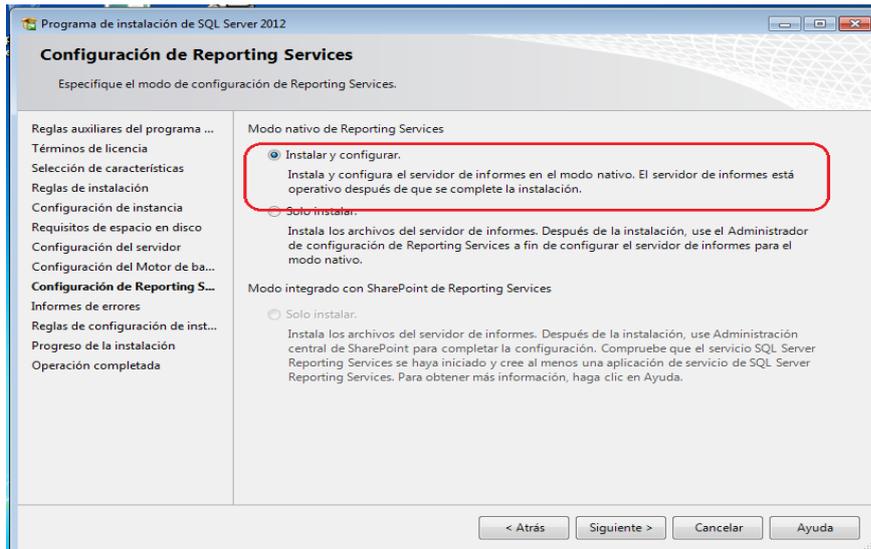
Seguir con la configuración de las cuentas que ejecutaran los distintos servicios incluidos (el motor del sgbd, reporting services y SQL Server browser). Recordar que SQL Server Browser es como un "agente" que publica en la red la existencia de la instancia de SQL Server. Por defecto viene deshabilitado.



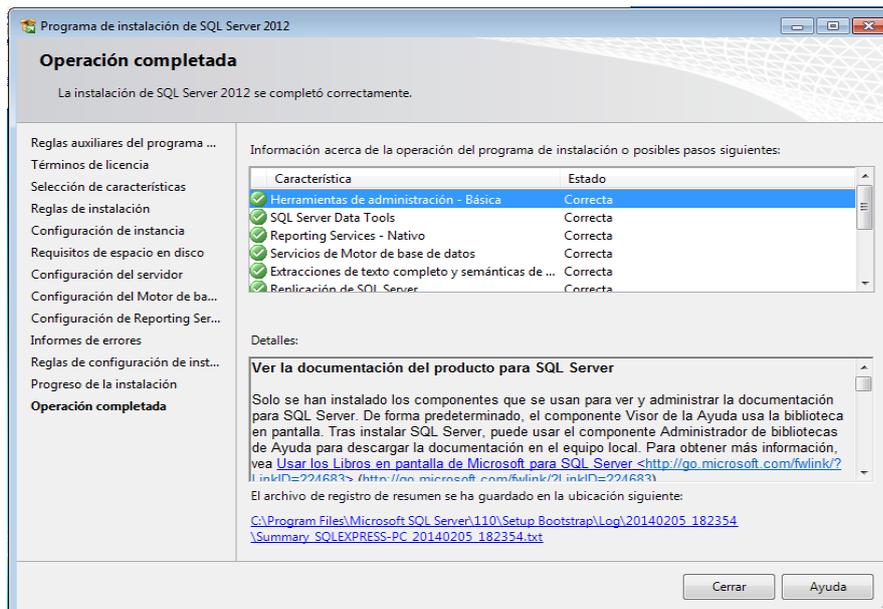
El siguiente paso consiste en elegir el modo de autenticación y los administradores de la base de datos. Elegir mixto si esta base de datos se necesita usar en la red local o autenticación de Windows si directamente es de uso local, a continuación elegir modo de autenticación de Windows y como administrador el usuario actual.



Uno de los últimos pasos es será elegir la configuración de Reporting Services. Para esta instalación simple basta con dejar la primera opción seleccionada, la que permite usarlo en modo nativo SIN integración con SharePoint services.



Esperar hasta que concluya la notificación de errores de Microsoft. Esperar una pantalla como la siguiente:



Finalmente reiniciar y listo ya está instalado. Si desea modificar cualquier configuración se puede cambiar las opciones generales si vamos al Menú Inicio > SQL Server 2012 > Herramientas de Configuración > Administración de configuración de SQL SERVER.

ANEXO II

**SCRIPT PARA CREAR LA BASE DE
DATOS PARA LAS PRUEBAS,
TRIGGERS, CHECK CONSTRAINT,
CAPTURAS DE LAS PRUEBAS –
MANAGE ENGINE**

SCRIPT PARA CREAR LA BASE DE DATOS PARA PARA LAS PRUEBAS DE CONSUMO DE RECURSOS Y NÚMERO DE ÓRDENES POR MINUTO

```
#OPCIONES EDITABLES#####
#NUMERO AL AZAR
proc RandomNumber {m M} {return [expr {int($m+rand()*($M+1-$m))}] }
#Función NURand
proc NURand { iConst x y C } {return [ expr {((([RandomNumber 0 $iConst] | [RandomNumber
$x $y]) + $C) % ($y - $x + 1)) + $x }}] }
#NOMBRE AL AZAR
proc randname { num } {
array set namearr { 0 BAR 1 OUGHT 2 ABLE 3 PRI 4 PRES 5 ESE 6 ANTI 7 CALLY 8 ATION 9
EING }
set name [ concat $namearr([ expr {( $num / 100 ) % 10 }])$namearr([ expr {( $num / 10 ) % 10
}])$namearr([ expr {( $num / 1 ) % 10 }]) ]
return $name
}
#TIMESTAMP
proc gettimestamp { } {
set tstamp [ clock format [ clock seconds ] -format %Y%m%d%H%M%S ]
return $tstamp
}
#KEYING TIME
proc keytime { keying } {
after [ expr {$keying * 1000} ]
return
}
#THINK TIME
proc thinktime { thinking } {
set thinkingtime [ expr {abs(round(log(rand())) * $thinking)} ]
after [ expr {$thinkingtime * 1000} ]
return
}
#CONEXION A POSTGRESQL
proc ConnectToPostgres { host port user password dbname } {
global tcl_platform
if {[catch {set lda [pg_connect -conninfo [list host = $host port = $port user = $user password =
$password dbname = $dbname ]]}] } {
puts stderr "Error, the database connection to $host could not be established"
```

```

set Ida "Failed"
} else {
if {$tcl_platform(platform) == "windows"} {
# Solución para el Bug #95 donde la primera conexión falla en Windows
catch {pg_disconnect $Ida}
set Ida [pg_connect -conninfo [list host = $host port = $port user = $user password = $password
dbname = $dbname ]]
}
pg_notice_handler $Ida puts
set result [ pg_exec $Ida "set CLIENT_MIN_MESSAGES TO 'ERROR'" ]
pg_result $result -clear
}
return $Ida
}
#NUEVA ORDEN
proc neword { Ida no_w_id w_id_input RAISEERROR ora_compatible } {
#Selecciona el district id al azar desde home warehouse donde el d_w_id = d_id
set no_d_id [ RandomNumber 1 10 ]
#Customer id seleccionado al azar donde c_d_id = d_id y c_w_id = w_id
set no_c_id [ RandomNumber 1 3000 ]
#Items ordenados al azar seleccionados desde 5 to 15
set ol_cnt [ RandomNumber 5 15 ]
#Fecha de la entrada de pedidos O_ENTRY_D generado por SUT
set date [ gettimestamp ]
if { $ora_compatible eq "true" } {
set result [pg_exec $Ida "exec
neword($no_w_id,$w_id_input,$no_d_id,$no_c_id,$ol_cnt,0,TO_TIMESTAMP($date,'YYYYMM
DDHH24MISS'))" ]
} else {
set result [pg_exec $Ida "select neword($no_w_id,$w_id_input,$no_d_id,$no_c_id,$ol_cnt,0)" ]
}
if {[pg_result $result -status] != "PGRES_TUPLES_OK"} {
if { $RAISEERROR } {
error "[pg_result $result -error]"
} else {
puts "New Order Procedure Error set RAISEERROR for Details"
}
} else {
puts "New Order: $no_w_id $w_id_input $no_d_id $no_c_id $ol_cnt 0 [ pg_result $result -list ]"
}
}

```

```

pg_result $result -clear
    }
}
#PAGO
proc payment { lda p_w_id w_id_input RAISEERROR ora_compatible } {
#El home warehouse id se mantiene igual por cada terminal
#Selecciona district id al azar desde home warehouse donde d_w_id = d_id
set p_d_id [ RandomNumber 1 10 ]
#Cliente seleccionado 60% del tiempo por nombre y 40% del tiempo por número
set x [ RandomNumber 1 100 ]
set y [ RandomNumber 1 100 ]
if { $x <= 85 } {
set p_c_d_id $p_d_id
set p_c_w_id $p_w_id
} else {
#usa una warehouse remota
set p_c_d_id [ RandomNumber 1 10 ]
set p_c_w_id [ RandomNumber 1 $w_id_input ]
while { ($p_c_w_id == $p_w_id) && ($w_id_input != 1) } {
set p_c_w_id [ RandomNumber 1 $w_id_input ]
}
}
set nrnd [ NURand 255 0 999 123 ]
set name [ randname $nrnd ]
set p_c_id [ RandomNumber 1 3000 ]
if { $y <= 60 } {
#usa nombre del cliente
#C_LAST es generado
set byname 1
} else {
#usa número del cliente
set byname 0
set name {}
}
#cantidad aleatoria 1-5000
set p_h_amount [ RandomNumber 1 5000 ]
#2.5.1.4 date selected from SUT
set h_date [ gettimestamp ]
#Transacción de pago

```

```

#cambiar siguiente para corregir los valores
if { $ora_compatible eq "true" } {
set result [pg_exec $lda "exec
payment($p_w_id,$p_d_id,$p_c_w_id,$p_c_d_id,$p_c_id,$byname,$p_h_amount,$name','0',0,
TO_TIMESTAMP($h_date,'YYYYMMDDHH24MISS'))" ]
} else {
set result [pg_exec $lda "select
payment($p_w_id,$p_d_id,$p_c_w_id,$p_c_d_id,$p_c_id,$byname,$p_h_amount,$name','0',0)
" ]
}
if {[pg_result $result -status] != "PGRES_TUPLES_OK"} {
if { $RAISEERROR } {
error "[pg_result $result -error]"
} else {
puts "Payment Procedure Error set RAISEERROR for Details"
}
} else {
puts "Payment: $p_w_id $p_d_id $p_c_w_id $p_c_d_id $p_c_id $byname $p_h_amount $name
0 0 [ pg_result $result -list ]"
pg_result $result -clear
}
}
#ESTADO_ORDEN
proc ostat { lda w_id RAISEERROR ora_compatible } {
#seleccionar district id alazar desde home warehouse donde d_w_id = d_id
set d_id [ RandomNumber 1 10 ]
set nrnd [ NURand 255 0 999 123 ]
set name [ randname $nrnd ]
set c_id [ RandomNumber 1 3000 ]
set y [ RandomNumber 1 100 ]
if { $y <= 60 } {
set byname 1
} else {
set byname 0
set name {}
}
if { $ora_compatible eq "true" } {
set result [pg_exec $lda "exec ostat($w_id,$d_id,$c_id,$byname,$name)"]
} else {

```

```

set result [pg_exec $lda "select * from ostat($w_id,$d_id,$c_id,$byname,$name) as (ol_i_id
NUMERIC, ol_supply_w_id NUMERIC, ol_quantity NUMERIC, ol_amount NUMERIC,
ol_delivery_d TIMESTAMP, out_os_c_id INTEGER, out_os_c_last VARCHAR, os_c_first
VARCHAR, os_c_middle VARCHAR, os_c_balance NUMERIC, os_o_id INTEGER, os_entdate
TIMESTAMP, os_o_carrier_id INTEGER) ]
}
if {[pg_result $result -status] != "PGRES_TUPLES_OK"} {
if { $RAISEERROR } {
error "[pg_result $result -error]"
} else {
puts "Order Status Procedure Error set RAISEERROR for Details"
}
} else {
puts "Order Status: $w_id $d_id $c_id $byname $name [ pg_result $result -list ]"
pg_result $result -clear
}
}
#ENTREGA
proc delivery { lda w_id RAISEERROR ora_compatible } {
set carrier_id [ RandomNumber 1 10 ]
set date [ gettimestamp ]
if { $ora_compatible eq "true" } {
set result [pg_exec $lda "exec
delivery($w_id,$carrier_id,TO_TIMESTAMP($date,'YYYYMMDDHH24MISS'))" ]
} else {
set result [pg_exec $lda "select delivery($w_id,$carrier_id)" ]
}
if {[pg_result $result -status] ni {"PGRES_TUPLES_OK" "PGRES_COMMAND_OK"}} {
if { $RAISEERROR } {
error "[pg_result $result -error]"
} else {
puts "Delivery Procedure Error set RAISEERROR for Details"
}
} else {
puts "Delivery: $w_id $carrier_id [ pg_result $result -list ]"
pg_result $result -clear
}
}
# NIVEL DE STOCK

```

```

proc slev { Ida w_id stock_level_d_id RAISEERROR ora_compatible } {
set threshold [ RandomNumber 10 20 ]
if { $ora_compatible eq "true" } {
set result [pg_exec $Ida "exec slev($w_id,$stock_level_d_id,$threshold)" ]
} else {
set result [pg_exec $Ida "select slev($w_id,$stock_level_d_id,$threshold)" ]
}
if {[pg_result $result -status] ni {"PGRES_TUPLES_OK" "PGRES_COMMAND_OK"}} {
if { $RAISEERROR } {
error "[pg_result $result -error]"
} else {
puts "Stock Level Procedure Error set RAISEERROR for Details"
}
} else {
puts "Stock Level: $w_id $stock_level_d_id $threshold [ pg_result $result -list ]"
pg_result $result -clear
}
}
#EJECUTAR TPC-C
set Ida [ ConnectToPostgres $host $port $user $password $db ]
if { $Ida eq "Failed" } {
error "error, the database connection to $host could not be established"
} else {
if { $ora_compatible eq "true" } {
set result [ pg_exec $Ida "exec SGBD_output.disable" ]
pg_result $result -clear
}
}
pg_select $Ida "select max(w_id) from warehouse" w_id_input_arr {
set w_id_input $w_id_input_arr(max)
}
# warehouse_id conjunto se mantiene constante para un terminal dado
set w_id [ RandomNumber 1 $w_id_input ]
pg_select $Ida "select max(d_id) from district" d_id_input_arr {
set d_id_input $d_id_input_arr(max)
}
set stock_level_d_id [ RandomNumber 1 $d_id_input ]
puts "Processing $total_iterations transactions without output suppressed..."
for {set it 0} {$it < $total_iterations} {incr it} {

```

```

if { [ tsv::get application abort ] } { break }
set choice [ RandomNumber 1 23 ]
if { $choice <= 10 } {
puts "new order"
if { $KEYANDTHINK } { keytime 18 }
neword $Ida $w_id $w_id_input $RAISEERROR $ora_compatible
if { $KEYANDTHINK } { thinktime 12 }
} elseif { $choice <= 20 } {
puts "payment"
if { $KEYANDTHINK } { keytime 3 }
payment $Ida $w_id $w_id_input $RAISEERROR $ora_compatible
if { $KEYANDTHINK } { thinktime 12 }
} elseif { $choice <= 21 } {
puts "delivery"
if { $KEYANDTHINK } { keytime 2 }
delivery $Ida $w_id $RAISEERROR $ora_compatible
if { $KEYANDTHINK } { thinktime 10 }
} elseif { $choice <= 22 } {
puts "stock level"
if { $KEYANDTHINK } { keytime 2 }
slev $Ida $w_id $stock_level_d_id $RAISEERROR $ora_compatible
if { $KEYANDTHINK } { thinktime 5 }
} elseif { $choice <= 23 } {
puts "order status"
if { $KEYANDTHINK } { keytime 2 }
ostat $Ida $w_id $RAISEERROR $ora_compatible
if { $KEYANDTHINK } { thinktime 5 }
}
}
pg_disconnect $Ida

```

SCRIPT PARA CREAR LA BASE DE DATOS PARA LAS PRUEBAS DE TIEMPO DE RESPUESTA Y TIEMPO DEL ASEGURAMIENTO DE LA INTEGRIDAD DE DATOS.

POSTGRESQL

```

CREATE DATABASE tdr1_pgs
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default

```

```
LC_COLLATE = 'Spanish_Spain.1252'  
LC_CTYPE = 'Spanish_Spain.1252'  
CONNECTION LIMIT = -1;
```

```
CREATE TABLE warehouse  
(  
  id_warehouse integer NOT NULL,  
  w_nombre character varying(64),  
  w_direccion1 character varying(64),  
  w_direccion2 character varying(64),  
  w_ciudad character varying(64),  
  w_estado character varying(64),  
  w_fecha_creacion date,  
  w_telefono character varying(64),  
  CONSTRAINT pk_warehouse PRIMARY KEY (id_warehouse)  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE warehouse  
  OWNER TO postgres;
```

```
CREATE TABLE distrito  
(  
  id_distrito integer NOT NULL,  
  id_warehouse integer,  
  d_nombre character varying(64),  
  d_direccion character varying(64),  
  d_calle1 character varying(64),  
  d_calle2 character varying(64),  
  d_barrio character varying(64),  
  d_fecha_creacion date,  
  d_capacidad integer,  
  CONSTRAINT pk_distrito PRIMARY KEY (id_distrito),  
  CONSTRAINT fk_distrito FOREIGN KEY (id_warehouse)  
    REFERENCES warehouse (id_warehouse) MATCH SIMPLE  
    ON UPDATE NO ACTION ON DELETE NO ACTION  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE distrito  
  OWNER TO postgres;
```

SCRIPT PARA MEDIR EL TIEMPO DE RESPUESTA EN LA INSERCIÓN DE REGISTROS

```
if (isset($_GET['ok'])){  
  $myServer = "MCORP\SQLEXPRESS, 1433";  
  $conex=array("Database"=>"tdr1_sql","UID"=>"yonyonet","PWD"=>"qwerty");  
  $conn = sqlsrv_connect ($myServer, $conex);  
  $cantidad = $_GET['reg_sql'];  
  
  $sql="exec insertarDistrito_Sql";
```

```

$start = (float) array_sum(explode(' ',microtime()));

if (!$conn) {
    echo "fallo";
}

for ($i = 1; $i <= $cantidad; $i++) {
    sqsrv_query($conn, $sql);
}

$end = (float) array_sum(explode(' ',microtime()));
//print "Processing time: ". sprintf("%.4f", ($end-$start))." seconds.";

$resultado = $end-$start;

}

if (isset($_GET['ok2'])){
    $conn= pg_connect("host=localhost port=5434 dbname=tdr1_pg user=postgres
password=sql1") or die ("Error de conexion. ". pg_last_error());
    $cantidad2 = $_GET['reg_postgres'];

    $sql="select insertar_distrito()";

    $start = (float) array_sum(explode(' ',microtime()));

    for ($i = 1; $i <= $cantidad2; $i++) {
        pg_query($conn,$sql);
    }

    $end = (float) array_sum(explode(' ',microtime()));
    //print "Processing time: ". sprintf("%.4f", ($end-$start))." seconds.";

    $resultado2 = $end-$start;

}

```

SCRIPT PARA MEDIR EL TIEMPO DE RESPUESTA EN LA CONSULTA DE REGISTROS

```

$sql = "exec ConsultarDistritos_Sql 1000";

$start = (float) array_sum(explode(' ', microtime()));

if (!$conn) {

    echo "fallo la conexion con la bd SQL";

}

```

```

$stmt = sqlsrv_query($conn, $sql);

if ((isset($resultado2)) || isset($cant)) {

?>

<?php

}

while ($row = sqlsrv_fetch_array($stmt, SQLSRV_FETCH_ASSOC)) {

    echo $row['id_distrito'] . ", " . $row['d_nombre'] . ", " . $row['d_direccion'] . ", " .
$row['d_calle1'] . ", " . $row['d_calle2'] . ", " . $row['d_barrio'] . ", " .
date_format($row['d_fecha_creacion'], 'YYYY,mm.dd') . ", " . $row['w_nombre'] .
$row['d_capacidad'] . "<br />";

}

sqlsrv_free_stmt($stmt);

$send = (float) array_sum(explode(' ', microtime()));

$resultado = $send - $start;

```

TRIGGER PARA PRUEBA DE TIEMPO DEL ASEGURAMIENTO DE LA INTEGRIDAD DE DATOS (POSTGRESQL)

```

CREATE OR REPLACE FUNCTION concatenarcalles_tri()
RETURNS trigger AS
$BODY$
BEGIN IF (TG_OP = 'INSERT') THEN
update distrito set d_direccion = d_calle1 || ' ' || d_calle2;
END IF;
RETURN NULL;
END;
$BODY$

```

CHECK PARA PRUEBA DE TIEMPO DEL ASEGURAMIENTO DE LA INTEGRIDAD DE DATOS (POSTGRESQL)

```
chk_fecha CHECK (d_fecha_creacion >= '2014-05-21'::date);
```

CAPTURAS DE LA HERRAMIENTA MANAGE ENGINE - PRUEBA CONSUMO DE RECURSOS

USO DE CPU

Utilización del CPU - Datos Agrupados para Monitor_Postgres_windows 8 de Wed Mar 26 18:28:04 COT 2014 a Wed Mar 26 18:51:10 COT 2014



Fecha	Hora	Valor en %
26-mar-2014	18:51	1
26-mar-2014	18:50	11
26-mar-2014	18:48	24
26-mar-2014	18:47	1
26-mar-2014	18:46	5
26-mar-2014	18:45	6
26-mar-2014	18:44	20
26-mar-2014	18:43	5
26-mar-2014	18:41	4
26-mar-2014	18:40	6
26-mar-2014	18:39	1
26-mar-2014	18:38	1
26-mar-2014	18:37	41
26-mar-2014	18:36	1
26-mar-2014	18:35	7
26-mar-2014	18:34	9
26-mar-2014	18:33	15
26-mar-2014	18:32	55
26-mar-2014	18:28	41

USO DE MEMORIA RAM



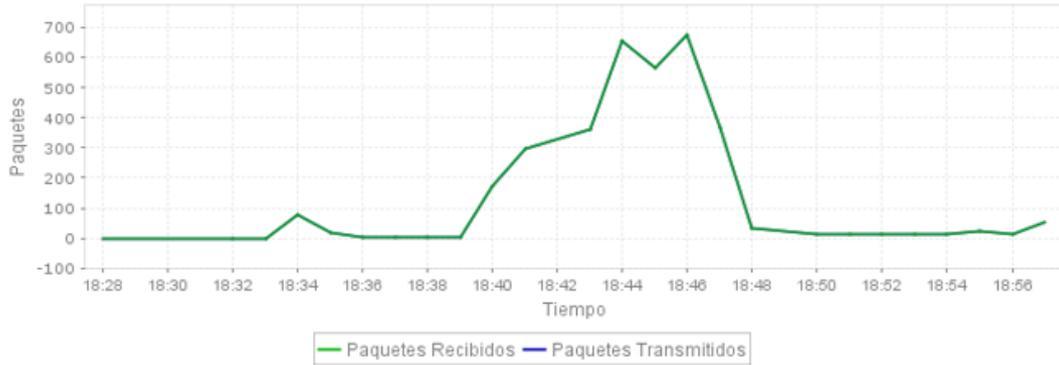
Fecha	Hora	Valor en %
26-mar-2014	18:53	31
26-mar-2014	18:52	31
26-mar-2014	18:51	31
26-mar-2014	18:50	30
26-mar-2014	18:48	30
26-mar-2014	18:47	30
26-mar-2014	18:46	30
26-mar-2014	18:45	30
26-mar-2014	18:44	30
26-mar-2014	18:43	30
26-mar-2014	18:41	30
26-mar-2014	18:40	30
26-mar-2014	18:39	29
26-mar-2014	18:38	29
26-mar-2014	18:37	30
26-mar-2014	18:36	29
26-mar-2014	18:35	29
26-mar-2014	18:34	29
26-mar-2014	18:33	28
26-mar-2014	18:32	29
26-mar-2014	18:28	29

USO DE RED

Interfaz de red - Detalles

Nombre del servidor	mcorp
Nombre de la interfaz	IntelR 82579V Gigabit Network Connection
Velocidad	100 Mbps

Paquetes



ANEXO III

ESTUDIO DE FACTIBILIDAD

FACTIBILIDAD TÉCNICA

1. HARDWARE

HARDWARE EXISTENTE

CANTIDAD	DESCRIPCION	ESTADO
1	Computadora de escritorio HP CORE I7 (servidor).	Optimo
>5	Computadora de escritorio HP CORE I7 (cliente)	Optimo
1	SWITCH	Optimo
1	Impresora Lexmark	Optimo
	Cables UTP directo	Optimo

HARDWARE REQUERIDO

CANTIDAD	DESCRIPCION	OBSERVACIONES
1	Computadora de escritorio	Debe tener la capacidad suficiente de procesamiento.
>5	Computadora de escritorio	
1	SWITCH	
1	Impresora	
	Cables UTP directo	

2. SOFTWARE

SOFTWARE EXISTENTE

NOMBRE	DESCRIPCION	No. LICENCIAS
Windows 7	Sistema Operativo	Todos los equipos tienen licencia.

SOFTWARE REQUERIDO

NOMBRE	DESCRIPCION	No. LICENCIAS
SQL SERVER	Base de datos	Licencia gratuita
Netbeans	Software de desarrollo	Software gratuito
Symfony	Framework de desarrollo.	Herramienta gratuita.
Windows 7	Sistema Operativo	Licencia necesaria.

3. PERSONAL TECNICO

PERSONAL TECNICO EXISTENTE

NOMBRE	FUNCIÓN
Johnny Parra	Jefe de proyecto, Programador
Johnny Parra, Mario Andrade	Diseñadores de base de datos, Analista, Programador
Mario Andrade	Programador

PERSONAL TECNICO REQUERIDO

NOMBRE	FUNCIÓN
Ing. Paul Paguay Soxo.	Asesor de Tesis
Johnny Parra	Jefe de proyecto, Programador
Johnny Parra, Mario Andrade	Diseñadores de base de datos, Analista, Programador
Mario Andrade	Programador

FACTIBILIDAD ECONÓMICA

1. COSTOS DE DESARROLLO

COSTO DE PERSONAL

CARGO	CANTIDAD	TIEMPO- MESES	COSTO- MESES	TOTAL
Jefe de proyecto y Programador	1	3	800,00	2400,00
Programador	1	3	800,00	2400,00
			Total	4800,00

COSTOS DE HARDWARE

CANTIDAD	DESCRIPCIÓN	COSTO UNITARIO	COSTO TOTAL
1	Impresora de tinta continua.	300,00	300,00

COSTOS DE SOFTWARE

CANTIDAD	DESCRIPCIÓN	COSTO UNITARIO	COSTO TOTAL
2	Netbeans	0,00	0,00
1	SQL SERVER Express Edition 2012	0,00	0,00
2	Windows 7	100,00	200,00
		Total	200,00

COSTOS DE SUMINISTROS

CANTIDAD	DESCRIPCIÓN	COSTO UNITARIO	COSTO TOTAL
1	Resma de papel	4,00	4,00
4	Frascos de tinta continua.	3,00	12,00
1	Suministros de oficina	15,00	15,00
3	CD's	0,40	1,20
		Total	32,20

2. COSTO DE INSTALAR EL SISTEMA

COSTO DE CAPACITACIÓN A USUARIOS

CANTIDAD	DESCRIPCIÓN	COSTO UNITARIO	COSTO TOTAL
5	Capacitación de usuarios	30,00	150,00
1	Personal durante la instalación	50,00	50,00
		Total	200,00

COSTO DE MANTENIMIENTO

CANTIDAD	TIEMPO-MESES	COSTO-MESES	COSTO TOTAL
Personal para dar mantenimiento	3	400,00	1200,00
		Total	1200,00

ANEXO IV

CÁLCULO DE ESTIMACIONES

CÁLCULO DE ESTIMACIONES

ILF: INTERNAL LOGIC FILE

Fichero lógico interno	DET	RET	complejidad
PERIODO	8	0	Baja
REQUISITO	3	0	Baja
CUMPLEREQUISITO	4	2	Baja
ESTUDIANTE	15	0	Baja
ASISTENCIA	9	1	Baja
MATRICULA	8	3	Baja
INSCRIPCIÓN	4	1	Baja
PARALELO	2	0	Baja
NIVEL	4	2	Baja
CURSO	2	0	Baja
MATERIAGRADO	4	2	Baja
MATERIAASIGNADA	6	2	Baja
SUBPERIODO	6	1	Baja
DICTADOMATERIA	5	4	Baja
MATERIA	6	0	Baja
EXAMENGRADO	6	2	Baja
DOCENTE	14	0	Baja
MATERIAPERIODO	3	2	Baja
HORA	4	0	Baja
EVALUACIÓN	9	2	Baja
ROL	3	0	Baja
HORARIOCLASE	5	2	Baja
DIA	2	0	Baja
MESEVALUACION	6	2	Baja

NOTA	3	0	Baja
USUARIO	10	1	Baja

❖ EI: ENTRADAS EXTERNAS

FTR	DET	COMPLEJIDAD
0	15	Baja
0	15	Baja
1	10	Baja
3	8	Alta
3	8	Alta
8	16	Alta
4	10	Alta
2	18	Alta
4	12	Alta
1	10	Baja
1	8	Baja
1	8	Baja
0	6	Baja
4	12	Alta
0	12	Baja
2	12	Media
0	12	Baja
0	6	Baja
4	8	Alta
4	6	Alta
4	16	Alta

0	6	Baja
0	6	Baja
0	6	Baja
4	18	Alta

❖ **EO: SALIDAS EXTERNAS**

No FTR	No DET	Complejidad
0	3	Baja
0	6	Baja
2	8	Media
0	3	Baja
1	3	Baja
2	6	Media
0	3	Baja
3	8	Media

❖ **EQ: CONSULTAS EXTERNAS**

ENTRADA		SALIDA	
DET	FTR	DET	FTR
3	3	4	3
3	3	10	3
3	1	5	1
3	2	8	2
3	2	4	2
3	2	4	2

3	2	11	2
3	2	15	2
3	2	13	2
3	1	12	1

C. Entrada	C. Salida	Complejidad
Baja	Media	Media
Baja	Baja	Baja
Media	Baja	Media
Media	Alta	Alta

- RESUMEN: CALCULO DE LOS PUNTOS DE FUNCIÓN

Parámetro	Complejidad	Número	Peso	Total
ILF	ALTA		15	
	MEDIO		10	
	BAJO	26	7	182
EIF	ALTA		10	
	MEDIO		7	
	BAJO		5	
EI	ALTA	12	6	72
	MEDIO	1	4	4
	BAJO	13	3	39
EO	ALTA		7	
	MEDIO	3	5	15
	BAJO	5	4	20

EQ	ALTA		6	
	MEDIO	6	4	24
	BAJO	4	3	12
	TOTAL puntos de función			368

Para sacar el valor del total de las líneas de código del sistema se hace uso de dos parámetros, los cuales se deben de multiplicar:

- Total puntos de función
- Valor proporcionado al elegir un determinado lenguaje de programación: 29 para el lenguaje orientado a objetos para el efecto.

- **SLOC= 368 x 29**
- **SLOC= 10672** líneas de código

A continuación se presenta algunas pantallas con el fin de proporcionar información de todos los parámetros que fueron ingresados en el programa como II para calcular el costo del proyecto y el tiempo que se necesita para ello:

- **Factor de ajuste**

base + Incr % = rating

Product: RELY DATA DOCU CPLX RUSE
base NOM HI VLO HI HI
Incr% 0% 0% 0% 0% 0%

Platform: TIME STOR EVOL
base NOM NOM NOM
Incr% 0% 0% 0%

Personnel: ACAP PCAP PCON APEX LTEX PLEX
base HI NOM HI NOM NOM LO
Incr% 0% 0% 0% 0% 0%

Project: TOOL SITE
base NOM NOM
Incr% 0% 0%

User: USR1 USR2
base NOM NOM
Incr% 0% 0%

EAF is also affected by Schedule
EAF: 0.96

OK Cancel Help

- Factor de esfuerzo

The dialog box titled "Scale Factors" contains the following data:

Factor	base	Incr%
Precedentedness	NOM	0%
Development Flexibility	VHI	0%
Architecture / risk resolution	LO	0%
Team cohesion	VHI	0%
Process maturity	HI	0%

Scale Factor : 14.60

- Puntos de función y líneas de código

The dialog box titled "SLOC Input Dialog - conduespoch" contains the following configuration:

- Sizing Method:** SLOC, Function Points, Adaptation and Reuse
- Breakage:** % of code thrown away due to requirements evolution and volatility. REVL: 0.05
- Module Size in Function Points:** Language: Object Oriented Default, Change Multiplier: 29
- Ratio Type:** Jones, David
- Calculation Method:** Using Table, Input Calculated Function Point

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Inputs	13	1	12	115
Outputs	5	3	0	35
Files	26	0	0	182
Interfaces	0	0	0	0
Queries	4	6	0	36
Total Unadjusted Function Points				368
Equivalent Total in SLOC				10672

- Resultados obtenidos

USC-COCOMO II.2000.4 - D:\Mis Documentos\Documentos\cocomo.est

File Edit View Parameters Calibrate Phase Maintenance Help

Project Name: **CONDUESPOCH** Scale Factor: 14.60 Schedule

Project Notes Development Model: Post Architecture

X	Module Name	Module Size	LABOR Rate (\$/month)	EAF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	conduespoch	F:10677	800.00	0.96	Object-Orient	35.8	34.5	309.0	27639.31	2.6	3.1	0.0

Total Lines of Code:	10677	Estimated	Effort Sched	PROD	COST	INST	Staff	RISK
Hours/PM:	152.00	Optimistic	27.6 10.2	386.3	22111.45	2.1	2.7	
		Most Likely	34.5 11.0	309.0	27639.31	2.6	3.1	0.0
		Pessimistic	43.2 11.8	247.2	34549.13	3.2	3.7	

ACT PM DEV: Actual Person Month Development: Read Only

ANEXO V

ANÁLISIS DE RIESGOS

ANÁLISIS DE RIESGOS

RIESGOS DEL PROYECTO			
ID	DESCRIPCIÓN	TIPO	CONSECUENCIA
R_01	No se cumple con el cronograma planificado	Proyecto	<ul style="list-style-type: none"> • Paro momentáneo del desarrollo del sistema. • Retraso en la entrega del proyecto. • No se puede concluir el proyecto.
R_02	Falta de explicitad del usuario en cuanto a sus necesidades.	Proyecto	<ul style="list-style-type: none"> • Inconsistencia de información en el desarrollo. • Redundancia de datos.
R_03	Incomprensión entre el equipo de trabajo.	Proyecto	<ul style="list-style-type: none"> • Pérdida de los avances del proyecto
R_04	El usuario cambia continuamente los requerimientos.	Proyecto	<ul style="list-style-type: none"> • Demora. • Incremento de costos.
R_05	Falta de compromiso por parte de los miembros del equipo de trabajo.	Proyecto	<ul style="list-style-type: none"> • Retraso del proyecto. • Incremento de costos.
RIESGOS TÉCNICOS			
R_06	Falta de experiencia con la plataforma de desarrollo.	Técnico	<ul style="list-style-type: none"> • del desarrollo del sistema, retraso del tiempo.
R_07	La plataforma de desarrollo presenta problemas de compatibilidad.	Técnico	<ul style="list-style-type: none"> • No se obtiene el resultado deseado.
R_08	No contar con el Hardware requerido.	Técnico	<ul style="list-style-type: none"> • Retraso. • Ineficiencia de respuestas de la Aplicación.
R_09	Diseño inadecuado de la base de datos.	Técnico	<ul style="list-style-type: none"> • Retraso del desarrollo del sistema, retraso del tiempo.

R_10	Diseño inadecuado de la red referente a la topología.	Técnico	<ul style="list-style-type: none"> • Dificultad en el manejo. • Demasiado tráfico en la transmisión de datos.
RIESGOS DE NEGOCIO			
R_11	La empresa quiebra.	Negocio	<ul style="list-style-type: none"> • El proyecto se suspende definitivamente.
R_12	Cambio de administrativos de la empresa.	Negocio	<ul style="list-style-type: none"> • El proyecto no se implementa por decisión de la nueva administración.

ANÁLISIS DE RIESGOS

ID RIESGO	DESCRIPCION	PROBABILIDAD			IMPACTO		EXPOSICION	
		PORCENTAJE	PROBABILIDAD	VALOR	IMPACTO	VALOR	EXPOSICION	VALOR
R_01	No se cumple con el cronograma planificado	30%	baja	1	moderado	2	Baja	2
R_02	Falta de explicitad del usuario en cuanto a sus necesidades.	45%	media	2	alto	3	Alta	6
R_03	Incomprensión entre el equipo de trabajo.	35%	media	2	moderado	2	Media	4
R_04	El usuario cambia continuamente los requerimientos.	30%	baja	1	moderado	2	Baja	2
R_05	Falta de compromiso por parte de los miembros del equipo de trabajo.	20%	baja	1	moderado	2	Baja	2
R_06	Falta de experiencia con la plataforma de desarrollo.	40%	media	2	alto	3	Media	3
R_07	La plataforma de desarrollo presenta problemas de compatibilidad.	30%	baja	1	alto	3	Media	3

R_08	No contar con el Hardware requerido.	30%	baja	1	alto	3	Media	3
R_09	Diseño inadecuado de la base de datos.	25%	baja	1	moderado	2	Baja	2
R_10	Diseño inadecuado de la red referente a la topología.	25%	baja	1	moderado	2	Baja	2
R_11	La empresa quiebra.	10%	baja	1	crítico	4	Baja	2
R_12	Cambio de administrativos de la empresa.	15%	baja	1	crítico	4	Media	4

ANÁLISIS DE RIESGOS				
ID	DESCRIPCION	EXPOSICION	VALOR	PRIORIDAD
R_02	Falta de explicitación del usuario en cuanto a sus necesidades.	Alta	6	1
R_03	Incomprensión entre el equipo de trabajo.	Media	6	2
R_01	No se cumple con el cronograma planificado	Baja	3	3
R_04	El usuario cambia continuamente los requerimientos.	Baja	3	3

R_07	La plataforma de desarrollo presenta problemas de compatibilidad.	Baja	3	3
R_08	No contar con el Hardware requerido.	Baja	3	3
R_09	Diseño inadecuado de la base de datos.	Baja	3	3
R_10	Diseño inadecuado de la red referente a la topología.	Baja	3	3
R_06	Falta de experiencia con la plataforma de desarrollo.	Baja	3	3
R_05	Falta de compromiso por parte de los miembros del equipo de trabajo.	Baja	3	4
R_12	Cambio de administrativos de la empresa.	Baja	3	4
R_11	La empresa quiebra.	Baja	3	4

ANEXO VI

**SPRINTS DEL SISTEMA,
PLANIFICACIÓN DEL PROYECTO,
DICCIONARIO DE DATOS**

SPRINTS DEL SISTEMA

- TAREAS DEL SPRINT 1**

TAREA DE INGENIERÍA	
Historia de Usuario: H1	
Número Tarea: 2	Nombre de la Tarea: Análisis de la información obtenida producto de las reuniones y delimitación del alcance del sistema.
Tipo de Tarea: Análisis de requerimientos	Puntos de Estimados: 6
Fecha de Inicio: 10/06/2014	Fecha de Fin: 12/06/2014
Responsables:	
Descripción:	
<ul style="list-style-type: none"> Luego de que se ha recopilado los requerimientos en las reuniones con el personal de la institución, se analizará que es lo que se va a realizar. Se preparará una propuesta dirigida a la institución con el objetivo de hacerles conocer que es lo que se planea llevar a cabo. 	

- SPRINT 2**

SPRINT 2	
Fecha Inicio:	13/06/2014
Fecha Fin:	01/07/2014
Descripción:	Creación de la base de datos Inscripción y visualización de estudiantes Creación de usuarios y roles
Esfuerzo estimado:	13 días
Responsable:	Johnny Parra, Mario Andrade

- HISTORIAS DE USUARIO DEL SPRINT 2**

HISTORIA DE USUARIO 2	
Número: H2	Nombre de la historia: Se creará la base de datos para uso del sistema académico

Usuario: Desarrollador	Iteración Asignada: 2
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como equipo de desarrollo se necesita crear la base de datos	
Observaciones:	

Historia de Usuario 2 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar la creación correcta de la base de datos	Verificar que se hayan creado todos los campos con los tipos de datos correctos. Probar el usuario propietario designado de la base de datos.

HISTORIA DE USUARIO 3	
Número: H3	Nombre de la historia: El sistema permitirá registrar un estudiante e inscribirlo en el periodo actual.
Usuario: Secretaria	Iteración Asignada: 2
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder inscribir a un estudiante en el sistema	
Observaciones: <ul style="list-style-type: none"> Si el estudiante ya fue previamente inscrito ya no será necesario volver a registrarlo en el sistema. 	

Historia de Usuario 3 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro e inscripción de estudiante	Probar registrar un estudiante que ya ha sido inscrito.

	<p>Probar buscar un estudiante que aún no ha sido registrado.</p> <p>Probar registrar un estudiante habiendo vencido el periodo de inscripciones.</p> <p>Probar ingresar datos no validos en el formulario de inscripción.</p>
--	--

HISTORIA DE USUARIO 4	
Número: H4	Nombre de la historia: El sistema permitirá modificar los datos de un estudiante que fue previamente inscrito
Usuario: Secretaria	Iteración Asignada: 2
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 17
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder modificar los datos de un estudiante	
Observaciones: <ul style="list-style-type: none"> • El rol Secretaria es el único ente autorizado en realizar esta funcionalidad • Solo se podrá modificar sus datos personales 	

Historia de Usuario 4 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar que se actualicen los datos del estudiante inscrito	<p>Probar que al momento de la búsqueda del estudiante a modificar lo realice correctamente.</p> <p>Probar que los cambios realizados fueron guardados correctamente.</p>

HISTORIA DE USUARIO 5	
Número: H5	Nombre de la historia: El sistema permitirá consultar todos los estudiantes inscritos en el periodo actual.
Usuario: Secretaria	Iteración Asignada: 2
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder visualizar los estudiantes previamente inscritos.	
Observaciones: <ul style="list-style-type: none"> Se listará haciendo uso de la paginación todos los estudiantes inscritos en el sistema 	

Historia de Usuario 5 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test consulta de estudiantes inscritos.	Probar que los estados de las inscripciones se muestren correctamente. Probar los enlaces de modificación de inscripción y requisitos. Probar como se visualiza la paginación.

HISTORIA DE USUARIO 6	
Número: H6	Nombre de la historia: El sistema permitirá crear usuarios
Usuario: Administrador del sistema	Iteración Asignada: 2
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder crear usuarios	
Observaciones: <ul style="list-style-type: none"> Se asignará a los usuarios sus respectivos roles, para su posterior ingreso al sistema 	

Historia de Usuario 6 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar el control de acceso	Prueba de control de acceso único del Administrador a realizar esta funcionalidad.
Test para verificar la creación de usuarios	<p>Prueba que se cree la clave de usuario encriptada</p> <p>Prueba que se cree con el rol asignado</p> <p>Prueba que luego pueda tener acceso al sistema</p> <p>Probar el registro de campos vacíos.</p> <p>Probar la autenticación del usuario según el rol que se le asigne.</p> <p>Probar actualizar el rol de usuario y verificar la redirección a la portada que le corresponde.</p> <p>Probar la actualización de campos con datos a campos vacíos.</p> <p>Probar cambiar de estado a los usuarios.</p> <p>Probar que el usuario inactivo no pueda autenticarse.</p> <p>Probar cambio de contraseña de usuario y autenticación con nuevas credenciales.</p> <p>Probar que los datos modificados se guarden exitosamente.</p>

- **TAREAS DEL SPRINT 2**

TAREA DE INGENIERÍA	
Historia de Usuario: H2	
Número Tarea: 3	Nombre de la Tarea: Diseñar la base de datos
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 13/06/2014	Fecha de Fin: 18/06/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se diseñará la base de datos tomando en cuenta los requerimientos planteados en las etapas iniciales • Se construirá su diseño conceptual. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H2	
Número Tarea: 4	Nombre de la Tarea: Creación de la base de datos
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 19/06/2014	Fecha de Fin: 20/06/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> Se creará las distintas tablas con sus respectivos atributos, tipos de datos y relaciones 	

TAREA DE INGENIERÍA	
Historia de Usuario: H3	
Número Tarea: 5	Nombre de la Tarea: Creación de patrones de rutas para registrar un estudiante e inscribirlo en el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 23/06/2014	Fecha de Fin: 23/06/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> Se creará las distintas rutas en el archivo de enrutamiento en relación al registro de un estudiante e inscripción en el periodo actual. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H3	
Número Tarea: 6	Nombre de la Tarea: Creación de vistas para registrar un estudiante e inscribirlo en el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 23/06/2014	Fecha de Fin: 23/06/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> Se creará las distintas plantillas que mostraran la información requerida respecto al registro de un estudiante e inscripción en el periodo actual. Se podrá hacer búsqueda de la inscripción de un estudiante dado algún identificador. Se mostrará la lista de requerimientos que aún le faltan cumplir al estudiante que se ha inscrito. Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H3	
Número Tarea: 7	Nombre de la Tarea: Creación de funciones y métodos para registrar un estudiante e inscribirlo en el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 24/06/2014	Fecha de Fin: 25/06/24
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para registrar un estudiante e inscribirlo en el periodo actual, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. • Se creará funciones que devuelvan la lista de requerimiento que aún faltan por cumplir a un estudiante registrado. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H4	
Número Tarea: 8	Nombre de la Tarea: Crear un patrón de ruta en el archivo de enrutamiento para acceder a modificar los datos del estudiante
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 26/06/2014	Fecha de Fin: 26/06/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una nueva ruta para acceder a modificar los datos del estudiante	

TAREA DE INGENIERÍA	
Historia de Usuario: H4	
Número Tarea: 9	Nombre de la Tarea: Crear métodos y funciones que permitan modificar los datos de un estudiante
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 26/06/2014	Fecha de Fin: 26/06/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará un método para buscar un determinado estudiante • Se creará una función que permita guardar los datos del estudiante que fue modificado 	

TAREA DE INGENIERÍA	
Historia de Usuario: H4	
Número Tarea: 10	Nombre de la Tarea: Crear la vista para presentar los datos del estudiante a modificar
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 26/06/2014	Fecha de Fin: 26/06/2014
Programador Responsable: Mario Andrade	
Descripción: Se obtendrá los datos y se presentará en la vista	

TAREA DE INGENIERÍA	
Historia de Usuario: H5	
Número Tarea: 11	Nombre de la Tarea: Creación de patrones de rutas para consultar todos los estudiantes inscritos en el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 27/06/2014	Fecha de Fin: 27/06/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación a la visualización del listado de los estudiantes inscritos en el periodo actual. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H5	
Número Tarea: 12	Nombre de la Tarea: Creación de vistas para visualizar todos los estudiantes inscritos en el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 27/06/2014	Fecha de Fin: 27/06/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará la plantilla que mostraran la información requerida respecto a la lista de estudiantes inscritos en el periodo actual. • Existirá enlaces que permitan consultar los requisitos de la inscripción y permitan modificar los datos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H5	
Número Tarea: 13	Nombre de la Tarea: Creación de funciones y métodos para visualizar todos los estudiantes inscritos en el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 27/06/2014	Fecha de Fin: 27/06/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará las funciones y métodos necesarios para visualizar todos los estudiantes inscritos en el periodo actual, entre ellos se encuentra métodos de consulta, búsqueda, validación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H6	
Número Tarea: 14	Nombre de la Tarea: Crear un patrón de ruta nueva en el archivo de enrutamiento para acceder a ingresar usuarios.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 30/06/2014	Fecha de Fin: 30/06/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una nueva ruta en la archivo de enrutamiento para acceder al registro de usuarios	

TAREA DE INGENIERÍA	
Historia de Usuario: H6	
Número Tarea: 15	Nombre de la Tarea: Crear la vista para mostrar el formulario de registro de usuarios.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 30/06/2014	Fecha de Fin: 30/06/2014
Programador Responsable: Johnny Parra	
Descripción: Se creará la vista con los campos de registro del nuevo usuario	

TAREA DE INGENIERÍA	
Historia de Usuario: H6	
Número Tarea: 16	Nombre de la Tarea: Crear métodos y funciones que permitan crear un nuevo usuario
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 01/07/2014	Fecha de Fin: 01/07/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> Se creará función para registrar un nuevo usuario en el sistema 	

- SPRINT 3**

SPRINT 3	
Fecha Inicio:	02/07/2014
Fecha Fin:	29/07/2014
Descripción:	Registro y visualización de matrículas Asignación de materias a docentes Asignación y visualización de carga horaria a docentes
Esfuerzo estimado:	20 días
Responsable:	Johnny Parra, Mario Andrade

- HISTORIAS DE USUARIO DEL SPRINT 3**

HISTORIA DE USUARIO 7	
Número: H7	Nombre de la historia: El sistema permitirá ver los requisitos que deben cumplir los estudiantes, previo al proceso de matriculación.
Usuario: Secretaria	Iteración Asignada: 3
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 17
Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)	

<p>Descripción:</p> <p>Como secretaria deseo poder ver los requisitos que deben cumplir los estudiantes previo al proceso de matriculación.</p>
<p>Observaciones:</p> <ul style="list-style-type: none"> EL usuario con rol Secretaria es el único que tiene acceso a esta funcionalidad.

Historia de Usuario 7 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test visualización de requisitos para la matriculación.	<p>Prueba mostrar si la lista no contiene elementos.</p> <p>Prueba visualizar elementos.</p>

HISTORIA DE USUARIO 8	
Número: H8	Nombre de la historia: El sistema permitirá matricular un estudiante en el sistema.
Usuario: Secretaria	Iteración Asignada: 3
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
<p>Descripción:</p> <p>Como secretaria deseo poder matricular un estudiante en el sistema.</p>	
<p>Observaciones:</p> <ul style="list-style-type: none"> EL usuario con rol Secretaria es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 8 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test matricula del estudiante en el sistema	<p>Prueba que esta previamente inscrito</p> <p>Prueba que cumpla todos los requisitos para el efecto</p> <p>Prueba que permita matricular una solo vez al estudiante en el sistema</p> <p>Prueba que se asigne las materias correspondientes a los su periodos activos</p> <p>Prueba que se asignen las notas parciales con valores por defecto</p> <p>Prueba que se asignen las asistencias con valores por defecto</p>

HISTORIA DE USUARIO 9	
Número: H9	Nombre de la historia: El sistema permitirá modificar los datos de una determinada matrícula.
Usuario: Secretaria	Iteración Asignada: 3
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder modificar los datos de una determinada matrícula.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Secretaria es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 9 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test modificar matrícula.	Probar cambiar de sección a la matrícula. Probar cambiar de curso.

HISTORIA DE USUARIO 10	
Número: H10	Nombre de la historia: El sistema permitirá visualizar la lista de estudiantes matriculados.
Usuario: Secretaria	Iteración Asignada: 3
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder visualizar la lista de estudiantes matriculados.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Secretaria es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 10 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para listar estudiantes matriculados	Probar que los estados de las matrículas se muestren correctamente.

	Probar los enlaces de modificación de matrícula Probar como se visualiza la paginación.
--	--

HISTORIA DE USUARIO 11	
Número: H11	Nombre de la historia: El sistema permitirá asignar materias a los docentes.
Usuario: Director Pedagógico	Iteración Asignada: 3
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como director/a pedagógico deseo poder asignar materias a los docentes.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Director Pedagógico es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 11 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de asignación de materias a docentes.	Probar asignar la materia 2 veces al docente en un mismo curso. Probar asignar una misma materia a otro docente en otro curso. Probar asignar la misma materia otro docente en el mismo curso. Probar modificar el docente a quien ha sido asignada la materia. Probar eliminar la asignación de materia a un docente.

HISTORIA DE USUARIO 12	
Número: H12	Nombre de la historia: El sistema permitirá asignar la carga horaria a un docente.
Usuario: Director Pedagógico	Iteración Asignada: 3
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 17
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como director/a pedagógico deseo poder asignar la carga horaria a un docente.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Director Pedagógico es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 12 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar la asignación de carga horaria al docente	Prueba que se carguen las materias únicamente del periodo actual. Prueba que no exista cruce de horarios

HISTORIA DE USUARIO 13	
Número: H13	Nombre de la historia: El sistema permitirá visualizar la carga horaria de un docente.
Usuario: Director Pedagógico	Iteración Asignada: 3
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como director/a pedagógico deseo poder visualizar la carga horaria de un docente.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Director Pedagógico es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 13 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de visualización de carga horaria	Probar la paginación de las listas. Probar los enlaces existentes.

HISTORIA DE USUARIO 14	
Número: H14	Nombre de la historia: El sistema permitirá ver el horario de clases de cada docente
Usuario: Docente	Iteración Asignada: 3
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder ver mi horario de clases.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Docente es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 14 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar que se muestre el horario del docente	Prueba que exista un enlace para que el docente pueda acceder a ver su horario Prueba que se presente en un formato legible para el docente

- TAREAS DEL SPRINT 3**

TAREA DE INGENIERÍA	
Historia de Usuario: H7	
Número Tarea: 17	Nombre de la Tarea: Creación de patrones de rutas para ver los requisitos que deben cumplir los estudiantes, previo al proceso de matriculación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 02/07/2014	Fecha de Fin: 02/07/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> Se creará la ruta en el archivo de enrutamiento en relación a la visualización de los requisitos que deben cumplir los estudiantes, previo al proceso de matriculación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H7	
Número Tarea: 18	Nombre de la Tarea: Creación de vistas para ver los requisitos que deben cumplir los estudiantes, previo al proceso de matriculación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 02/07/2014	Fecha de Fin: 02/07/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> Se creará la plantilla que mostrará la información requerida respecto a los requisitos que deben cumplir los estudiantes, previo al proceso de matriculación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H7	
Número Tarea: 19	Nombre de la Tarea: Creación de funciones y métodos para ver los requisitos que deben cumplir los estudiantes, previo al proceso de matriculación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 02/07/2014	Fecha de Fin: 02/07/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará las funciones y métodos necesarios para ver la lista de los requisitos que deben cumplir los estudiantes, previo al proceso de matriculación, entre ellos se encuentra métodos de consulta, validación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H8	
Número Tarea: 20	Nombre de la Tarea: Crear los patrones de rutas que permitan matricular un estudiante
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 03/07/2014	Fecha de Fin: 03/07/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> Se creará una ruta para que permita acceder a buscar un estudiante en el sistema Se creará una ruta para que permita acceder a matricular un estudiante 	

TAREA DE INGENIERÍA	
Historia de Usuario: H8	
Número Tarea: 21	Nombre de la Tarea: Crear las funciones y métodos necesarios para matricular un estudiante
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 07/07/2014	Fecha de Fin: 11/07/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará un método para buscar al estudiante en el sistema • Se creará un método para verificar si un estudiante ya fue matriculado anteriormente • Se creará un método para verificar si el estudiante cumple con los requisitos necesarios para proceder a la matrícula • Se creará función para registrar la matrícula de un estudiante 	

TAREA DE INGENIERÍA	
Historia de Usuario: H8	
Número Tarea: 22	Nombre de la Tarea: Crear las distintas vistas que permitan matricular un estudiante
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 03/07/2014	Fecha de Fin: 04/07/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará una vista que permita buscar un estudiante en el sistema. • Se creará una vista que contenga los requisitos que cumple o no el estudiante • Se creará una vista que permita registrar una nueva matrícula 	

TAREA DE INGENIERÍA	
Historia de Usuario: H9	
Número Tarea: 23	Nombre de la Tarea: Creación de patrones de rutas para modificar los datos de una determinada matrícula.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 14/07/2014	Fecha de Fin: 14/07/2014
Programador Responsable: Johnny Parra	

Descripción:

- Se creará la ruta en el archivo de enrutamiento en relación a la modificación de los datos de una determinada matrícula.

TAREA DE INGENIERÍA**Historia de Usuario: H9****Número Tarea: 24****Nombre de la Tarea:** Creación de vistas para modificar los datos de una determinada matrícula.**Tipo de Tarea:** Desarrollo**Puntos de Estimados: 5****Fecha de Inicio:** 14/07/2014**Fecha de Fin:** 14/07/2014**Programador Responsable:** Johnny Parra**Descripción:**

- Se creará la plantilla que mostrará la información requerida respecto a la modificación de los datos de una determinada matrícula, por secciones y cursos.
- Se mostrará un mensaje al terminar la transacción.

TAREA DE INGENIERÍA**Historia de Usuario: H9****Número Tarea: 25****Nombre de la Tarea:** Creación de funciones y métodos para modificar los datos de una determinada matrícula.**Tipo de Tarea:** Desarrollo**Puntos de Estimados: 8****Fecha de Inicio:** 14/07/2014**Fecha de Fin:** 14/07/2014**Programador Responsable:** Johnny Parra**Descripción:**

- Se creará las funciones y métodos necesarios para modificar los datos de una determinada matrícula, entre ellos se encuentra métodos de consulta, búsqueda, validación y modificación de datos.
- Se creará funciones que devuelvan la lista de matrículas divididos por secciones y cursos.

TAREA DE INGENIERÍA	
Historia de Usuario: H10	
Número Tarea: 26	Nombre de la Tarea: Crear una nueva ruta en el archivo de enrutamiento que permita ver la lista de estudiantes matriculados
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 15/07/2014	Fecha de Fin: 15/07/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una nueva ruta para acceder a ver la lista de estudiantes matriculados	
TAREA DE INGENIERÍA	
Historia de Usuario: H10	
Número Tarea: 27	Nombre de la Tarea: Crear los métodos que devuelva la lista de estudiantes matriculados en el periodo actual
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 15/07/2014	Fecha de Fin: 15/07/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará un método que devuelva la lista de cursos • Se hará uso de paginación para presentar la lista de estudiantes encontrados • Se creará un método que permite buscar de estudiantes matriculados en el periodo actual 	

TAREA DE INGENIERÍA	
Historia de Usuario: H10	
Número Tarea: 28	Nombre de la Tarea: Crear una vista para presentar los datos de los estudiantes matriculados
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 15/07/2014	Fecha de Fin: 15/07/2014
Programador Responsable: Mario Andrade	
Descripción: La vista contendrá la lista de cursos, y los estudiantes matriculados mediante paginación.	

TAREA DE INGENIERÍA	
Historia de Usuario: H11	
Número Tarea: 29	Nombre de la Tarea: Creación de patrones de rutas para asignar materias a los docentes.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 16/07/2014	Fecha de Fin: 16/07/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará la ruta en el archivo de enrutamiento en relación a la asignación de materias a los docentes. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H11	
Número Tarea: 30	Nombre de la Tarea: Creación de vistas para asignar materias a los docentes.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 16/07/2014	Fecha de Fin: 17/07/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará las distintas plantillas que mostraran la información requerida respecto a la asignación de materias a los docentes. Se mostrará la lista de materias y para quien han sido asignadas. Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H11	
Número Tarea: 31	Nombre de la Tarea: Creación de funciones y métodos para asignar materias a los docentes.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 18/07/2014	Fecha de Fin: 21/07/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará las funciones y métodos necesarios asignar materias a los docentes, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos, actualización y eliminación. Se creará funciones que devuelvan la lista de materias y a quien han sido asignadas. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H12	
Número Tarea: 32	Nombre de la Tarea: Crear una nueva ruta en el archivo de enrutamiento para asignar horarios a los docentes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 22/07/2014	Fecha de Fin: 22/07/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una nueva ruta que permita registrar el horario de los docentes.	

TAREA DE INGENIERÍA	
Historia de Usuario: H12	
Número Tarea: 33	Nombre de la Tarea: Crear al vista que permita asignar la carga horario al docente
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 22/07/2014	Fecha de Fin: 23/07/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una vista que presente el formulario de registro de horarios de docentes	

TAREA DE INGENIERÍA	
Historia de Usuario: H12	
Número Tarea: 34	Nombre de la Tarea: Crear los métodos y funciones que permitan asignar la carga horaria a los docentes.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 24/07/2014	Fecha de Fin: 24/07/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará un método que devuelva la lista de materias de un docente junto con los días y horas que previamente fueron asignadas por defecto. • Se creará una función que permita guardar las materias que a los docentes le fueron asignadas. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H13	
Número Tarea: 35	Nombre de la Tarea: Creación de patrones de rutas para visualizar la carga horaria de un docente.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 25/07/2014	Fecha de Fin: 25/07/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará la ruta en el archivo de enrutamiento en relación a la visualización de la carga horaria de los docentes. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H13	
Número Tarea: 36	Nombre de la Tarea: Creación de vistas para visualizar la carga horaria de un docente.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 25/07/2014	Fecha de Fin: 25/07/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará la plantilla que mostrará la información requerida respecto a la visualización de la carga horaria de los docentes. Se mostrará la lista con la información requerida y enlaces que permitan la modificación de los mismos. Se mostrará un mensaje en caso de realizar alguna transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H13	
Número Tarea: 37	Nombre de la Tarea: Creación de funciones y métodos para visualizar la carga horaria de un docente.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 25/07/2014	Fecha de Fin: 25/07/2014
Programador Responsable: Johnny Parra	
Descripción:	

- Se creará las funciones y métodos necesarios para visualizar la carga horaria de un docente, entre ellos se encuentra métodos de consulta, validación y modificación de datos.

TAREA DE INGENIERÍA	
Historia de Usuario: H14	
Número Tarea: 38	Nombre de la Tarea: Crear una nueva ruta en el archivo de enrutamiento para acceder al horario de clases del docente.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 28/07/2014	Fecha de Fin: 28/07/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una ruta que permita acceder al horario de clases de un docente que esté autenticado en el sistema.	

TAREA DE INGENIERÍA	
Historia de Usuario: H14	
Número Tarea: 39	Nombre de la Tarea: Crear la vista para presentar el horario de clases de un determinado docente
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 28/07/2014	Fecha de Fin: 28/07/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una vista, la cual contendrá el horario de clases, especificada cada materia con su día, hora y curso	

TAREA DE INGENIERÍA	
Historia de Usuario: H14	
Número Tarea: 40	Nombre de la Tarea: Crear método que devuelva la carga horaria de un determinado docente
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 29/07/2014	Fecha de Fin: 29/07/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará una consulta que devuelva el horario de clases de un docente. 	

- **SPRINT 4**

SPRINT 4	
Fecha Inicio:	30/07/2014
Fecha Fin:	02/09/2014
Descripción:	Registro y visualización de asistencias Registro y visualización de notas
Esfuerzo estimado:	25 días
Responsable:	Johnny Parra, Mario Andrade

- **HISTORIAS DE USUARIO DEL SPRINT 4**

HISTORIA DE USUARIO 15	
Número: H15	Nombre de la historia: El sistema permitirá registrar las asistencias de los estudiantes por materias y cursos.
Usuario: Inspector	Iteración Asignada: 4
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como inspector deseo poder registrar las asistencias de los estudiantes por materias y cursos.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Inspector es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 15 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro de asistencia.	Probar ingresar cantidades negativas y caracteres que no correspondan.

HISTORIA DE USUARIO 16	
Número: H16	Nombre de la historia: El sistema permitirá visualizar las asistencias de los estudiantes por materias y cursos.
Usuario: Inspector	Iteración Asignada: 4
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como inspector deseo poder visualizar las asistencias de los estudiantes por materias y cursos.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Inspector es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 16 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar la visualización de asistencia de estudiantes	Prueba que se carguen los cursos activos en el periodo actual Prueba que se carguen las materias correspondientes a los subperiodos activos Prueba que se visualice la lista de estudiantes con sus valores de asistencias correctas

HISTORIA DE USUARIO 17	
Número: H17	Nombre de la historia: El sistema permitirá registrar las notas parciales por curso y materia.
Usuario: Docente	Iteración Asignada: 4
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 18
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder registrar las notas parciales por curso y materia.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Docente es el único que tiene acceso a esta funcionalidad. El registro de notas estará controlado por intervalos de fechas. 	

Historia de Usuario 17 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro de notas parciales.	Probar insertar cantidades negativas o caracteres especiales. Probar ingresar un parcial habiendo vencido el tiempo de su registro.

HISTORIA DE USUARIO 18	
Número: H18	Nombre de la historia: El sistema permitirá visualizar las notas parciales por curso y materia.
Usuario: Docente	Iteración Asignada: 4
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder visualizar las notas parciales por curso y materia.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Docente es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 18 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para visualizar las notas parciales	Prueba que se cargue la lista de notas parciales a consultar. Prueba que se carguen todas las materias activas en el periodo actual Prueba que se visualice correctamente las notas parciales en base a la materia y curso correspondiente

HISTORIA DE USUARIO 19	
Número: H19	Nombre de la historia: El sistema permitirá registrar las notas de suspenso por curso y materia.
Usuario: Docente	Iteración Asignada: 4
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder registrar las notas de suspenso por curso y materia.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Docente es el único que tiene acceso a esta funcionalidad. • El registro de las notas de suspenso estará controlado por intervalos de fechas. 	

Historia de Usuario 19 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test registro de suspenso	Probar insertar cantidades negativas o caracteres especiales. Probar ingresar un parcial habiendo vencido el tiempo de su registro. Probar actualización de notas y equivalencias de aprobación.

HISTORIA DE USUARIO 20	
Número: H20	Nombre de la historia: El sistema permitirá visualizar las notas de suspenso por curso y materia.
Usuario: Docente	Iteración Asignada: 4
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Media (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder visualizar las notas de suspenso por curso y materia.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Docente es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 20 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar la visualización de notas de suspenso	Prueba que se carguen los cursos y sus materias de los subperiodos activos. Prueba que se visualice la lista de estudiantes con sus notas de suspenso correspondiente

HISTORIA DE USUARIO 21	
Número: H21	Nombre de la historia: El sistema permitirá registrar las notas de grado por curso y materia.
Usuario: Secretaria	Iteración Asignada: 4
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder registrar las notas de grado por curso y materia.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Docente es el único que tiene acceso a esta funcionalidad. • El registro de las notas de grado estará controlado por intervalos de fechas. 	

Historia de Usuario 21 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test registro de notas de grado	Probar insertar cantidades negativas o caracteres especiales. Probar ingresar una nota habiendo vencido el tiempo de su registro. Probar actualización de notas y equivalencias de aprobación.

HISTORIA DE USUARIO 22	
Número: H22	Nombre de la historia: El sistema permitirá visualizar las notas de grado por curso y materia.
Usuario: Secretaria	Iteración Asignada: 4
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder visualizar las notas de grado por curso y materia.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Secretaria es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 22 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar la visualización de notas de grado	Prueba que se carguen los cursos activos en el periodo actual Prueba que se carguen las materias de grado correspondientes Prueba que se visualicen todos los estudiantes con sus respectivas notas correspondientes.

- TAREAS DEL SPRINT 4**

TAREA DE INGENIERÍA	
Historia de Usuario: H15	
Número Tarea: 41	Nombre de la Tarea: Creación de patrones de rutas para registrar las asistencias de los estudiantes por materias y cursos.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 30/07/2014	Fecha de Fin: 30/07/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> Se creará las distintas rutas en el archivo de enrutamiento en relación al registro de las asistencias de los estudiantes por materias y cursos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H15	
Número Tarea: 42	Nombre de la Tarea: Creación de vistas para registrar las asistencias de los estudiantes por materias y cursos.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 30/07/2014	Fecha de Fin: 31/07/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará la plantilla que mostrará la información requerida respecto al registro de las asistencias de los estudiantes por materias y cursos. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H15	
Número Tarea: 43	Nombre de la Tarea: Creación de funciones y métodos para registrar las asistencias de los estudiantes por materias y cursos.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 01/08/2014	Fecha de Fin: 05/08/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para registrar las asistencias de los estudiantes por materias y cursos, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. • Se creará la función que presente la lista de estudiantes con los campos a ser llenados, por materia y curso. • Se creará la función que calcule el promedio de asistencia por estudiante para mostrarlo en la vista. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H16	
Número Tarea: 44	Nombre de la Tarea: Crear una nueva ruta en el archivo de enrutamiento que permita acceder a ver las asistencias de los estudiantes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 06/08/2014	Fecha de Fin: 06/08/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una nueva ruta que permita acceder a ver las asistencias de estudiantes	

TAREA DE INGENIERÍA	
Historia de Usuario: H16	
Número Tarea: 45	Nombre de la Tarea: Crear los métodos necesarios para poder ver las asistencias de los estudiantes.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 06/08/2014	Fecha de Fin: 06/08/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará un método que devuelva todos los cursos • Se creará un método que devuelva la lista de materias que correspondan a los subperiodos activos, luego de haber seleccionado un curso. • Se creará un método que liste los registros de asistencias los estudiantes de una determinada materia. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H16	
Número Tarea: 46	Nombre de la Tarea: Crear una vista para presentar la asistencia de los estudiantes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 06/08/2014	Fecha de Fin: 06/08/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una vista, la cual contendrá la lista de cursos y materias mediante los cuales se podrá acceder a la lista de estudiantes con sus respectivas asistencias	

TAREA DE INGENIERÍA	
Historia de Usuario: H17	
Número Tarea: 47	Nombre de la Tarea: Creación de patrones de rutas para registrar las notas parciales por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 07/08/2014	Fecha de Fin: 07/08/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al registro de las notas parciales por curso y materia. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H17	
Número Tarea: 48	Nombre de la Tarea: Creación de vistas registrar las notas parciales por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 07/08/2014	Fecha de Fin: 08/08/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto al registro de las notas parciales por curso y materia. • Se mostrará la lista de estudiantes con los campos a ser llenados correspondientes a las notas del parcial. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H17	
Número Tarea: 49	Nombre de la Tarea: Creación de funciones y métodos para registrar las notas parciales por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 11/08/2014	Fecha de Fin: 14/08/2014
Programador Responsable: Johnny Parra	
Descripción:	

- Se creará las funciones y métodos para registrar las notas parciales por curso y materia, entre ellos se encuentra métodos de consulta, validación y registro de datos.
- Se creará la función que devuelvan la lista de estudiantes con los campos a ser llenados correspondientes a las notas del parcial.

TAREA DE INGENIERÍA	
Historia de Usuario: H18	
Número Tarea: 50	Nombre de la Tarea: Crear una nueva ruta en el archivo de enrutamiento para acceder a ver las notas parciales
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 15/08/2014	Fecha de Fin: 15/08/2014
Programador Responsable: Johnny Parra	
Descripción: Se creará una nueva ruta que permita visualizar las notas parciales de cada materia en un curso determinado.	

TAREA DE INGENIERÍA	
Historia de Usuario: H18	
Número Tarea: 51	Nombre de la Tarea: Crear los métodos que permitan visualizar las notas parciales
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 15/08/2014	Fecha de Fin: 15/08/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará un método que lista todos los parciales que existen en el sistema • Se creará un método que devuelvan las materias que pertenecen al docente autenticado • Se creará un método que devuelva la lista de estudiante con sus respectivas notas 	

TAREA DE INGENIERÍA	
Historia de Usuario: H18	
Número Tarea: 52	Nombre de la Tarea: Crear una vista en la cual se pueda presentar las notas parciales de los estudiantes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 15/08/2014	Fecha de Fin: 15/08/2014
Programador Responsable: Johnny Parra	
Descripción: Se creará una vista con el fin de poder mostrar los datos enviados a la misma	

TAREA DE INGENIERÍA	
Historia de Usuario: H19	
Número Tarea: 53	Nombre de la Tarea: Creación de patrones de rutas para registrar las notas de suspenso por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 18/08/2014	Fecha de Fin: 18/08/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al registro de las notas de suspenso por curso y materia. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H19	
Número Tarea: 54	Nombre de la Tarea: Creación de vistas para registrar las notas de suspenso por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 18/08/2014	Fecha de Fin: 19/08/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto al registro de las notas de suspenso por curso y materia. • Se mostrará la lista de estudiantes con el campo para el ingreso de la nota de suspenso. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H19	
Número Tarea: 55	Nombre de la Tarea: Creación de funciones y métodos para registrar las notas de suspenso por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 20/08/2014	Fecha de Fin: 22/08/2014
Programador Responsable: Johnny Parra	
Descripción:	

- Se creará las funciones y métodos necesarios para registrar las notas de suspenso por curso y materia, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos.
- Se creará la función que devuelvan la lista de estudiantes con el campo para el ingreso de la nota de suspenso.
- Se creará la función que actualice las respectivas equivalencias y promedios luego de haber ingresado la nota de suspenso.

TAREA DE INGENIERÍA	
Historia de Usuario: H20	
Número Tarea: 56	Nombre de la Tarea: Crear una nueva ruta en el archivo de enrutamiento para acceder a las notas de suspensos
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 25/08/2014	Fecha de Fin: 25/08/2014
Programador Responsable: Johnny Parra	
Descripción: Se creará una nueva ruta que permita acceder a la vista que contiene la lista de estudiantes con las notas de suspensos.	

TAREA DE INGENIERÍA	
Historia de Usuario: H20	
Número Tarea: 57	Nombre de la Tarea: Crear métodos que devuelva la lista de estudiantes con sus notas de supletorios.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 25/08/2014	Fecha de Fin: 25/08/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará método para obtener la lista de estudiantes que únicamente están en suspenso • Se enviará a guardar las notas y se actualizará la equivalencia de la nota promedio final 	

TAREA DE INGENIERÍA	
Historia de Usuario: H20	
Número Tarea: 58	Nombre de la Tarea: Crear una vista para presentar las notas de suspenso de los estudiantes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 25/08/2014	Fecha de Fin: 25/08/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará una vista para poder visualizar las notas de supletorios de todos los estudiantes • Se controlará el ingreso de notas mediante el intervalo de fechas que dura el periodo académico 	

TAREA DE INGENIERÍA	
Historia de Usuario: H21	
Número Tarea: 59	Nombre de la Tarea: Creación de patrones de rutas para registrar las notas de grado por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 26/08/2014	Fecha de Fin: 26/08/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al registro de las notas de grado por curso y materia. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H21	
Número Tarea: 60	Nombre de la Tarea: Creación de vistas para registrar las notas de grado por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 26/08/2014	Fecha de Fin: 27/08/2014
Programador Responsable: Mario Andrade	
Descripción:	

- Se creará las distintas plantillas que mostraran la información requerida respecto al registro de las notas de grado por curso y materia.
- Se mostrara la lista de materias de grado activadas para el periodo actual.
- Se mostrará la lista de estudiantes con el campo para el ingreso de la nota de grado.
- Se mostrará un mensaje al terminar la transacción.

TAREA DE INGENIERÍA	
Historia de Usuario: H21	
Número Tarea: 61	Nombre de la Tarea: Creación de funciones y métodos para registrar las notas de grado por curso y materia.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 28/08/2014	Fecha de Fin: 01/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para registrar las notas de grado por curso y materia, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. • Se creará la función que devuelvan la lista de estudiantes con el campo para el ingreso de la nota de grado. • Se creará la función que devuelva la lista de materias de grado activadas para el periodo actual. • Se creará la función que actualice las respectivas equivalencias y promedios luego de haber ingresado la nota de suspenso. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H22	
Número Tarea: 62	Nombre de la Tarea: Crear una nueva ruta en el archivo de enrutamiento para acceder a las notas de grado
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 02/09/2014	Fecha de Fin: 02/09/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una nueva ruta que permita acceder a la vista que contiene la lista de estudiantes con las notas de grado.	

TAREA DE INGENIERÍA	
Historia de Usuario: H22	
Número Tarea: 63	Nombre de la Tarea: Crear métodos que devuelva la lista de estudiantes con sus notas de grado.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 02/09/2014	Fecha de Fin: 02/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará método para obtener la lista de estudiantes con sus notas de las materias de grado • Se enviará a guardar las notas de grado. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H22	
Número Tarea: 64	Nombre de la Tarea: Crear una vista para presentar las notas de grado de los estudiantes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 02/09/2014	Fecha de Fin: 02/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará una vista para poder visualizar las notas de grado de todos los estudiantes 	

- **SPRINT 5**

SPRINT 5	
Fecha Inicio:	03/09/2014
Fecha Fin:	23/09/2014
Descripción:	Implementación de seguridad de acceso al sistema Desarrollo del módulo de administración general del sistema
Esfuerzo estimado:	15 días
Responsable:	Johnny Parra, Mario Andrade

- HISTORIAS DE USUARIO SPRINT 5

HISTORIA DE USUARIO 23	
Número: H23	Nombre de la historia: El sistema permitirá autenticar a los usuarios registrados en el sistema
Usuario: Desarrollador	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 20
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como equipo de desarrollo deseo poder realizar la parte de autenticación del sistema.	
Observaciones: <ul style="list-style-type: none"> • Se creará un firewall para manejar la seguridad de acceso en el sistema 	

Historia de Usuario 23 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de autenticación al sistema.	No pueden autenticarse los usuarios no registrados en el sistema. No pueden autenticarse los usuarios registrados cuyo estado no se encuentre activado.

HISTORIA DE USUARIO 24	
Número: H24	Nombre de la historia: El sistema permitirá hacer uso de perfiles, para tener acceso al mismo
Usuario: Desarrollador	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como equipo de desarrollo deseo poder manejar perfiles de usuario.	
Observaciones: <ul style="list-style-type: none"> • Se hará uso de perfiles de usuarios para el acceso al sistema 	

Historia de Usuario 24 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de perfiles de usuario.	De acuerdo al perfil de usuario se mostrará la portada que tiene asociada. Ante un mal ingreso en las credenciales de autenticación regresa al formulario de login.

HISTORIA DE USUARIO 25	
Número: H25	Nombre de la historia: El sistema permitirá que cada usuario autenticado pueda modificar sus datos personales
Usuario: Usuario del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como usuario registrado en el sistema deseo poder modificar mis datos de acceso al mismo.	
Observaciones: <ul style="list-style-type: none"> Cada usuario tendrá la facultad de modificar sus datos personales, incluida su clave de acceso al sistema. 	

Historia de Usuario 25 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de actualización de datos de acceso al sistema.	Probar si se actualizan los datos del usuario. En caso de no modificar la contraseña debe permanecer la anterior.

HISTORIA DE USUARIO 26	
Número: H26	Nombre de la historia: El sistema permitirá registrar un nuevo periodo académico
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder ingresar un nuevo periodo académico.	
Observaciones: <ul style="list-style-type: none"> • Se registrará un nuevo periodo académico solo si el anterior ya está desactivado • Se establecerá intervalos de fechas para las inscripciones, matrículas y del propio periodo académico 	

Historia de Usuario 26 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro de nuevo periodo	Probar ingresar campos vacíos. Probar ingresar un nuevo periodo, habiendo un periodo con estado activo.

HISTORIA DE USUARIO 27	
Número: H27	Nombre de la historia: El sistema permitirá modificar el periodo académico actual
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder modificar el actual periodo académico.	
Observaciones: <ul style="list-style-type: none"> • Se podrá modificar el nombre, estado, los intervalos de fechas para las inscripciones, matrículas y del propio periodo académico 	

Historia de Usuario 27 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de modificación de periodo	Probar ingresar campos vacíos.

HISTORIA DE USUARIO 28	
Número: H28	Nombre de la historia: El sistema permitirá ingresar, modificar y listar los parciales a evaluar en un periodo académico.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder ingresar, modificar y listar los parciales a evaluar en un periodo académico.	
Observaciones:	

Historia de Usuario 28 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro de parciales.	Probar ingreso de campos vacíos.
Test de modificación de parciales.	Probar ingreso de campos vacíos.

HISTORIA DE USUARIO 29	
Número: H29	Nombre de la historia: El sistema permitirá establecer el rango de fechas, mediante las cuales se podrá controlar el ingreso de notas por parte de los docentes
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder establecer el rango de fechas en que se podrán ingresar cada una de las evaluaciones parciales.	
Observaciones: <ul style="list-style-type: none"> • Tener en cuenta cada una de las fechas que se designan para ingresar las notas 	

Historia de Usuario 29 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de rango de fechas para parciales	Probar que según el intervalo de fechas el enlace para registrar la nota de un parcial no se muestra.

HISTORIA DE USUARIO 30	
Número: H30	Nombre de la historia: El sistema permitirá registrar los días, horas para posteriormente asignar la carga horaria a docentes.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 18
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción:	

Como administrador del sistema deseo poder establecer los días y las horas que serán utilizadas para la asignación de la carga horaria de docentes.
Observaciones:

Historia de Usuario 30 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para verificar el ingreso de días	Prueba que no permita ingresar días duplicados Prueba que se guarden correctamente los datos
Test para verificar el ingreso de horas	Prueba que no permita ingresar horas duplicadas Prueba que se guarden correctamente los datos

HISTORIA DE USUARIO 31	
Número: H31	Nombre de la historia: El sistema permitirá ingresar y modificar los subperiodos que contendrá el periodo actual.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder ingresar y modificar los subperiodos que contendrá el periodo actual.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. • En reglamento actual de la Institución se ha establecido que existen 3 subperiodos. • Al inicio del Periodo Académico actual deben estar activos el primer y último subperiodo. 	

Historia de Usuario 31 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro de subperiodo	<p>Probar cambiar los estados de los subperiodos.</p> <p>Probar los mensajes de la aplicación al activar múltiples periodos o desactivar todos los subperiodos.</p>

HISTORIA DE USUARIO 32	
Número: H32	Nombre de la historia: El sistema permitirá ingresar, modificar y listar las materias.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 18
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder ingresar, modificar y listar las materias en el sistema.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 32 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para el ingreso de materias	<p>Prueba que salgan todos los campos de la entidad materia para su registro</p> <p>Prueba que se guarden los cambios en la base de datos luego de su modificación</p> <p>Prueba que carguen en la vista todas las materias del sistema con sus datos</p> <p>Prueba que controle los campos obligatorios</p>

HISTORIA DE USUARIO 33	
Número: H33	Nombre de la historia: El sistema permitirá modificar de manera simultánea todas las materias previamente registradas en el sistema.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder modificar de manera simultánea todas las materias previamente registradas en el sistema.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 33 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de modificación de todas las materias	Probar que los datos modificados se guarden exitosamente. Probar actualizar el nombre de la materia dejando el campo vacío (no debe permitirse ingresar el campo nombre como vacío).

HISTORIA DE USUARIO 34	
Número: H34	Nombre de la historia: El sistema permitirá ingresar y modificar las materias de grado que serán evaluadas en el periodo actual.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 18
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder ingresar y modificar las materias de grado que serán evaluadas en el periodo actual.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. • Según el reglamento de la Institución se ingresaran las materias de grado para el periodo actual. 	

Historia de Usuario 34 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para el ingreso de materias de grado	Prueba que salgan todos los campos de la entidad materia de grado para su registro Prueba que se guarden los cambios en la base de datos luego de su modificación Prueba que carguen en la vista todas las materias de grado del sistema correspondiente al periodo actual. Prueba que controle los campos obligatorios

HISTORIA DE USUARIO 35	
Número: H35	Nombre de la historia: El sistema permitirá asignar y modificar las materias a los distintos subperiodos previamente registrados.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 16
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder asignar y modificar las materias a los distintos subperiodos previamente registrados.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. • Según el reglamento de la Institución se asignarán las materias al respectivo subperiodo dentro del periodo actual. 	

Historia de Usuario 35 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de modificación de asignación de materia a subperiodo.	Probar que los datos modificados se guarden exitosamente.

HISTORIA DE USUARIO 36	
Número: H36	Nombre de la historia: El sistema permitirá ingresar, modificar y listar: paralelos, niveles y cursos.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder ingresar, modificar y listar: paralelos, niveles y cursos.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 36 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test para el ingreso de paralelos	Prueba que salgan todos los campos de la entidad paralelo para su registro Prueba que se guarden los cambios en la base de datos luego de su modificación Prueba que carguen en la vista todos los paralelos del sistema con sus datos Prueba que controle los campos obligatorios
Test para el ingreso de niveles	Prueba que salgan todos los campos de la entidad nivel para su registro Prueba que se guarden los cambios en la base de datos luego de su modificación Prueba que carguen en la vista todos los niveles del sistema con sus datos Prueba que controle los campos obligatorios
Test para el ingreso de cursos	Prueba que salgan todos los campos de la entidad curso para su registro Prueba que se guarden los cambios en la base de datos luego de su modificación Prueba que carguen en la vista todos los cursos del sistema con sus datos Prueba que controle los campos obligatorios

HISTORIA DE USUARIO 37	
Número: H37	Nombre de la historia: El sistema permitirá ingresar, modificar y listar los requisitos que serán tomados en cuenta para el proceso de matriculación.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 19
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder ingresar, modificar y listar los requisitos que serán tomados en cuenta para el proceso de matriculación.	
Observaciones: <ul style="list-style-type: none"> • EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. • Según el reglamento de la Institución se registrarán los requisitos para el periodo actual. 	

Historia de Usuario 37 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro y actualización de requisitos.	Probar el registro de campos vacíos. Probar la actualización de campos con datos a campos vacíos. Probar cambiar de estado a los requisitos. Probar que los datos modificados se guarden exitosamente.

HISTORIA DE USUARIO 38	
Número: H38	Nombre de la historia: El sistema permitirá modificar de manera simultánea todos los requisitos previamente registrados
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 17
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como administrador del sistema deseo poder modificar de manera simultánea todos los requisitos previamente registrados.	
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad. 	

Historia de Usuario 38 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de actualización de requisitos.	Probar el registro de campos vacíos. Probar la actualización de campos con datos a campos vacíos. Probar cambiar de estado a los requisitos. Probar que los datos modificados se guarden exitosamente.

HISTORIA DE USUARIO 39	
Número: H39	Nombre de la historia: El sistema permitirá ingresar, modificar y listar los roles en el sistema.
Usuario: Administrador del sistema	Iteración Asignada: 5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 18
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción:	

Como administrador del sistema deseo poder ingresar, modificar y listar los roles en el sistema.
Observaciones: <ul style="list-style-type: none"> EL usuario con rol Administrador es el único que tiene acceso a esta funcionalidad.

Historia de Usuario 39 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de registro, actualización y listado de roles.	Probar el registro de campos vacíos. Probar la actualización de campos con datos a campos vacíos. Probar que los datos modificados se guarden exitosamente.

- TAREAS DEL SPRINT 5**

TAREA DE INGENIERÍA	
Historia de Usuario: H23	
Número Tarea: 65	Nombre de la Tarea: Creación de patrones de rutas para la autenticación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 03/09/2014	Fecha de Fin: 03/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> Se creará las distintas rutas en el archivo de enrutamiento en relación a los roles de usuario que vaya a manejar la aplicación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H23	
Número Tarea: 66	Nombre de la Tarea: Creación de la plantilla de login y funciones asociadas.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 03/09/2014	Fecha de Fin: 03/09/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará la plantilla para la autenticación además de las distintas funciones para la validación de datos, se configurara el tipo de codificación de contraseña y usuario.	

TAREA DE INGENIERÍA	
Historia de Usuario: H23	
Número Tarea: 67	Nombre de la Tarea: Creación del firewall para la aplicación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 03/09/2014	Fecha de Fin: 03/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> Se creará en el archivo respectivo las reglas de control de acceso necesarias a modo de firewall para controlar el acceso a la aplicación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H23	
Número Tarea: 68	Nombre de la Tarea: Creación de proveedores de usuarios.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 03/09/2014	Fecha de Fin: 03/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> Se configurará los archivos necesarios para establecer la creación de los usuarios a partir de la entidad seleccionada, también se crean los métodos que son requeridos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H24	
Número Tarea: 69	Nombre de la Tarea: Creación de patrones de rutas para perfiles de usuario.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 04/09/2014	Fecha de Fin: 04/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> Se creará las distintas rutas en el archivo de enrutamiento en relación a los perfiles de usuario que vaya a manejar la aplicación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H24	
Número Tarea: 70	Nombre de la Tarea: Creación de vistas de perfiles de usuario.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 04/09/2014	Fecha de Fin: 04/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto a los perfiles de usuario, en este caso existen 5 perfiles. • La plantilla deberá contar con un enlace para regresar a la página anterior. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H24	
Número Tarea: 71	Nombre de la Tarea: Creación de funciones y métodos para perfiles de usuario.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 04/09/2014	Fecha de Fin: 04/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para el manejo de perfiles de usuario, entre ellos métodos de búsqueda, selección, validaciones. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H25	
Número Tarea: 72	Nombre de la Tarea: Creación de patrones de rutas para la actualización de perfiles de usuario.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación a la actualización de los perfiles de usuario que vaya a manejar la aplicación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H25	
Número Tarea: 73	Nombre de la Tarea: Creación de vistas para la actualización de perfiles de usuario.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto a los perfiles de usuario, en este caso existen 5 perfiles. • La plantilla deberá contar con un enlace para regresar a la página anterior. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H25	
Número Tarea: 74	Nombre de la Tarea: Creación de funciones y métodos para la actualización de perfiles de usuario.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para el manejo de perfiles de usuario, entre ellos métodos de búsqueda, selección, validaciones. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H26	
Número Tarea: 75	Nombre de la Tarea: Creación de patrones de rutas para registrar nuevo periodo.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al registro de un nuevo periodo. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H26	
Número Tarea: 76	Nombre de la Tarea: Creación de vistas para registrar nuevo periodo.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto a la información de registro de un nuevo periodo. • Se mostrará un mensaje si se trata de ingresar un periodo habiendo ya un periodo activo. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H26	
Número Tarea: 77	Nombre de la Tarea: Creación de funciones y métodos para registrar nuevo periodo.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios que permitan registrar un nuevo periodo, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H27	
Número Tarea: 78	Nombre de la Tarea: Creación de patrones de rutas para modificar periodo.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación a la modificación de datos de un periodo. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H27	
Número Tarea: 79	Nombre de la Tarea: Creación de vistas para modificar un periodo.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto a la modificación de datos de un nuevo periodo. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H27	
Número Tarea: 80	Nombre de la Tarea: Creación de funciones y métodos para registrar nuevo periodo.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 05/09/2014	Fecha de Fin: 05/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para modificar un periodo, entre ellos se encuentra métodos de consulta, búsqueda, validación y actualización de datos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H28	
Número Tarea: 81	Nombre de la Tarea: Creación de patrones de rutas para registrar los parciales del periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 08/09/2014	Fecha de Fin: 08/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al registro de los parciales del periodo actual. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H28	
Número Tarea: 82	Nombre de la Tarea: Creación de vistas para registrar los parciales del periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 08/09/2014	Fecha de Fin: 08/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto al registro los parciales del periodo actual. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H28	
Número Tarea: 83	Nombre de la Tarea: Creación de funciones y métodos para registrar los parciales del periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 08/09/2014	Fecha de Fin: 08/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para registrar los parciales del periodo actual, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H29	
Número Tarea: 84	Nombre de la Tarea: Creación de patrones de rutas para establecer el rango de fechas para registro de parciales.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 09/09/2014	Fecha de Fin: 09/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al establecimiento de rangos de fechas para registro de parciales. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H29	
Número Tarea: 85	Nombre de la Tarea: Creación de vistas para establecer el rango de fechas para registro de parciales.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 09/09/2014	Fecha de Fin: 09/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto al establecimiento de rangos de fechas para registro de parciales del periodo actual. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H29	
Número Tarea: 86	Nombre de la Tarea: Creación de funciones y métodos para establecer el rango de fechas para registro de parciales.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 10
Fecha de Inicio: 09/09/2014	Fecha de Fin: 09/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para establecer el rango de fechas para el registro de los parciales del periodo actual, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H30	
Número Tarea: 87	Nombre de la Tarea: Crear los distintos patrones de ruta para el ingreso de días y horas
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 10/09/2014	Fecha de Fin: 10/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará en el archivo de enrutamiento los patrones de ruta necesarias para acceder al ingreso de días y horas. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H30	
Número Tarea: 88	Nombre de la Tarea: Crear las funciones que permita ingresar los días y horas
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 10/09/2014	Fecha de Fin: 10/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará una función que permita registrar los días • Se creará una función que permita registrar las horas 	

TAREA DE INGENIERÍA	
Historia de Usuario: H30	
Número Tarea: 89	Nombre de la Tarea: Crear las vistas necesarias que permitan el registrar los días y las horas
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 10/09/2014	Fecha de Fin: 10/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará una vista que permita registrar los días • Se creará una vista que permita registrar las horas 	

TAREA DE INGENIERÍA	
Historia de Usuario: H31	
Número Tarea: 90	Nombre de la Tarea: Creación de patrones de rutas para ingresar y modificar los subperiodos que contendrá el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 11/09/2014	Fecha de Fin: 11/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al registro y modificación de los subperiodos que contendrá el periodo actual. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H31	
Número Tarea: 91	Nombre de la Tarea: Creación de vistas para ingresar y modificar los subperiodos que contendrá el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 11/09/2014	Fecha de Fin: 11/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto al registro de los subperiodos que contendrá el periodo actual. • Se creará las distintas plantillas que mostraran la información requerida respecto a la modificación de los subperiodos que contendrá el periodo actual. • Se mostrará la lista de subperiodos registrados. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H31	
Número Tarea: 92	Nombre de la Tarea: Creación de funciones y métodos para ingresar y modificar los subperiodos que contendrá el periodo actual.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 11/09/2014	Fecha de Fin: 11/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para registrar y modificar los subperiodos que contendrá el periodo actual., entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. • Se creará funciones que devuelvan la lista de subperiodos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H32	
Número Tarea: 93	Nombre de la Tarea: Crear los patrones de rutas que permitan administrar las materias
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 12/09/2014	Fecha de Fin: 12/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará una rutas para el registro, modificación y listado de materias 	

TAREA DE INGENIERÍA	
Historia de Usuario: H32	
Número Tarea: 94	Nombre de la Tarea: Crear las funciones y métodos para el ingreso, modificación y listado de materias
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 12/09/2014	Fecha de Fin: 12/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará funciones para el registro, modificación de materias • Se creará métodos para consultar y listar materias 	

TAREA DE INGENIERÍA	
Historia de Usuario: H32	
Número Tarea: 95	Nombre de la Tarea: Crear las vistas para el registro, modificación y listado de materias
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 12/09/2014	Fecha de Fin: 12/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará vistas para el registro, modificación y listado de materias 	

TAREA DE INGENIERÍA	
Historia de Usuario: H33	
Número Tarea: 96	Nombre de la Tarea: Creación de patrones de rutas para modificar de manera simultánea todas las materias previamente registradas en el sistema.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 2
Fecha de Inicio: 15/09/2014	Fecha de Fin: 15/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación a la modificación de manera simultánea todas las materias previamente registradas en el sistema. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H33	
Número Tarea: 97	Nombre de la Tarea: Creación de vistas para modificar de manera simultánea todas las materias previamente registradas en el sistema.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 15/09/2014	Fecha de Fin: 15/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto a la modificación de manera simultánea todas las materias previamente registradas en el sistema. • Se mostrará la lista total de materias activas para la modificación. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H33	
Número Tarea: 98	Nombre de la Tarea: Creación de funciones y métodos para modificar de manera simultánea todas las materias previamente registradas en el sistema.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 15/09/2014	Fecha de Fin: 15/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para modificar de manera simultánea todas las materias, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. • Se creará funciones que devuelvan la lista de materias activas registradas. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H34	
Número Tarea: 99	Nombre de la Tarea: Crear los patrones de rutas que permitan administrar las materias de grado
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 16/09/2014	Fecha de Fin: 16/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará una rutas para el registro, modificación de materias de grado 	

TAREA DE INGENIERÍA	
Historia de Usuario: H34	
Número Tarea: 100	Nombre de la Tarea: Crear las funciones y métodos para el ingreso y modificación de materias de grado.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 16/09/2014	Fecha de Fin: 16/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará funciones y métodos para el registro, modificación de materias de grado 	

TAREA DE INGENIERÍA	
Historia de Usuario: H34	
Número Tarea: 101	Nombre de la Tarea: Crear las vistas para el registro, modificación de materias de grado
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 16/09/2014	Fecha de Fin: 16/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará vistas para el registro, modificación de materias de grado 	

TAREA DE INGENIERÍA	
Historia de Usuario: H35	
Número Tarea: 102	Nombre de la Tarea: Creación de patrones de rutas para asignar y modificar las materias a los distintos subperiodos previamente registrados.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 17/09/2014	Fecha de Fin: 17/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación a la asignación y modificación de las materias a los distintos subperiodos previamente registrados. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H35	
Número Tarea: 103	Nombre de la Tarea: Creación de vistas para asignar y modificar las materias a los distintos subperiodos previamente registrados.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 17/09/2014	Fecha de Fin: 17/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto a la asignación y modificación de las materias a los distintos subperiodos previamente registrados. • Se mostrará la lista total de materias activas con la información del periodo al que pertenecen. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H35	
Número Tarea: 104	Nombre de la Tarea: Creación de funciones y métodos para asignar y modificar las materias a los distintos subperiodos previamente registrados.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 17/09/2014	Fecha de Fin: 17/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para asignar y modificar las materias a los distintos subperiodos, entre ellos se encuentra métodos de consulta, búsqueda, validación y registro de datos. • Se creará funciones que devuelvan la lista total de materias activas con la información del periodo al que pertenecen. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H36	
Número Tarea: 105	Nombre de la Tarea: Crear los patrones de rutas que permitan administrar los paralelos, niveles y cursos
Tipo de Tarea: Desarrollo	Puntos de Estimados: 4
Fecha de Inicio: 18/09/2014	Fecha de Fin: 18/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará una rutas para el registro, modificación y listado de paralelos • Se creará una rutas para el registro, modificación y listado de niveles • Se creará una rutas para el registro, modificación y listado de cursos 	

TAREA DE INGENIERÍA	
Historia de Usuario: H36	
Número Tarea: 106	Nombre de la Tarea: Crear las funciones y métodos para la administración de paralelos, niveles y cursos.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 18/09/2014	Fecha de Fin: 18/09/2014

Programador Responsable: Johnny Parra
Descripción: <ul style="list-style-type: none"> • Se creará funciones y métodos para el registro, modificación y listado de paralelos • Se creará funciones y métodos para el registro, modificación y listado de niveles • Se creará funciones y métodos para el registro, modificación y listado de cursos

TAREA DE INGENIERÍA	
Historia de Usuario: H36	
Número Tarea: 107	Nombre de la Tarea: Crear las vistas para la administración de paralelos, niveles y cursos.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 18/09/2014	Fecha de Fin: 18/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> • Se creará vistas para el registro, modificación y listado de paralelos • Se creará vistas para el registro, modificación y listado de niveles • Se creará vistas para el registro, modificación y listado de cursos. 	
TAREA DE INGENIERÍA	
Historia de Usuario: H37	
Número Tarea: 108	Nombre de la Tarea: Creación de patrones de rutas para ingresar, modificar y listar los requisitos que serán tomados en cuenta para el proceso de matriculación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 4
Fecha de Inicio: 19/09/2014	Fecha de Fin: 19/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas rutas en el archivo de enrutamiento en relación al ingreso, modificación y visualización del listado de los requisitos que serán tomados en cuenta para el proceso de matriculación. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H37	
Número Tarea: 109	Nombre de la Tarea: Creación de vistas para ingresar, modificar y listar los requisitos que serán tomados en cuenta para el proceso de matriculación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 19/09/2014	Fecha de Fin: 19/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las distintas plantillas que mostraran la información requerida respecto al ingreso de los requisitos que serán tomados en cuenta para el proceso de matriculación. • Se creará las distintas plantillas que mostraran la información requerida respecto a la modificación de los requisitos que serán tomados en cuenta para el proceso de matriculación. • Se creará las distintas plantillas que mostraran la información requerida respecto a la visualización del listado de los requisitos que serán tomados en cuenta para el proceso de matriculación. • Se mostrará la lista total de requisitos activos. • Se mostrará un mensaje al terminar la transacción. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H37	
Número Tarea: 110	Nombre de la Tarea: Creación de funciones y métodos para ingresar los requisitos que serán tomados en cuenta para el proceso de matriculación.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 19/09/2014	Fecha de Fin: 19/09/2014
Programador Responsable: Mario Andrade	
Descripción: <ul style="list-style-type: none"> • Se creará las funciones y métodos necesarios para ingresar los requisitos que serán tomados en cuenta para el proceso de matriculación. • Se creará las funciones y métodos necesarios para modificar los requisitos que serán tomados en cuenta para el proceso de matriculación. • Se creará las funciones y métodos necesarios para mostrar la lista de los requisitos que serán tomados en cuenta para el proceso de matriculación. • Se creará funciones que devuelvan la lista total de requisitos activos. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H38	
Número Tarea: 111	Nombre de la Tarea: Crear patrones de rutas para acceder a modificar todos los requisitos
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 22/09/2014	Fecha de Fin: 22/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará las rutas para visualizar y modificar todos los requisitos registrados en el sistema 	

TAREA DE INGENIERÍA	
Historia de Usuario: H38	
Número Tarea: 112	Nombre de la Tarea: Crear funciones y métodos para modificar todos los requisitos
Tipo de Tarea: Desarrollo	Puntos de Estimados: 5
Fecha de Inicio: 22/09/2014	Fecha de Fin: 22/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará los métodos para buscar los requisitos registrados en el sistema • Se creará funciones para actualizar todos los requisitos de manera simultánea 	

TAREA DE INGENIERÍA	
Historia de Usuario: H38	
Número Tarea: 113	Nombre de la Tarea: Crear vista que permita contener todos los requisitos a modificar
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 22/09/2014	Fecha de Fin: 22/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará una vista que contenga todos los requisitos registrados previamente en el sistema 	

TAREA DE INGENIERÍA	
Historia de Usuario: H39	
Número Tarea: 114	Nombre de la Tarea: Crear patrones de rutas para la administración de los roles
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 23/09/2014	Fecha de Fin: 23/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se crearán rutas que permitan acceder al ingreso, modificación y listado de roles. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H39	
Número Tarea: 115	Nombre de la Tarea: Crear las funciones y métodos para la administración de los roles.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 23/09/2014	Fecha de Fin: 23/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará métodos para buscar y listar roles • Se creará funciones para el ingreso, modificación y listado de roles. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H39	
Número Tarea: 116	Nombre de la Tarea: Crear las vistas para el ingreso, modificación y listado de roles.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 23/09/2014	Fecha de Fin: 23/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará vistas para el ingreso, modificación y listado de roles 	

- **SPRINT 6**

SPRINT 6	
Fecha Inicio:	24/09/2014
Fecha Fin:	26/09/2014
Descripción:	Reportes
Esfuerzo estimado:	4 días
Responsable:	Johnny Parra, Mario Andrade

- **HISTORIAS DE USUARIO DEL SPRINT 6**

HISTORIA DE USUARIO 40	
Número: H40	Nombre de la historia: El sistema permitirá generar un reporte de los estudiantes en suspenso
Usuario: Docente	Iteración Asignada: 6
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 17
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder generar un reporte de los estudiantes que se quedaron en suspenso.	
Observaciones: <ul style="list-style-type: none"> • El reporte será generado en formato PDF 	

Historia de Usuario 40 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de generación de reporte estudiantes en suspenso.	Probar la generación de reportes con la lista de estudiantes vacía. Probar que los datos calculados y reflejados en el reporte sean los correctos. Probar imprimir el reporte. Probar la visualización del reporte desde varios navegadores.

HISTORIA DE USUARIO 41	
Número: H41	Nombre de la historia: El sistema permitirá generar un reporte de las notas parciales de un determinado mes y una materia
Usuario: Docente	Iteración Asignada: 6
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 17
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder generar un reporte de cada una de las notas parciales.	
Observaciones: <ul style="list-style-type: none"> El reporte será generado en formato PDF 	

Historia de Usuario 41 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de generación de reporte de las notas parciales de un determinado mes y una materia.	Probar la generación de reportes con la lista de estudiantes vacía. Probar que los datos calculados y reflejados en el reporte sean los correctos. Probar imprimir el reporte. Probar la visualización del reporte desde varios navegadores.

HISTORIA DE USUARIO 42	
Número: H42	Nombre de la historia: El sistema permitirá generar un reporte general de calificaciones de las notas parciales.
Usuario: Docente	Iteración Asignada: 6
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 18
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como docente deseo poder generar un reporte general de calificaciones de las notas parciales.	

Observaciones:

- El reporte será generado en formato PDF.

Historia de Usuario 42 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de generación de reporte general.	Probar la generación de reportes con la lista de estudiantes vacía. Probar que los datos calculados y reflejados en el reporte sean los correctos. Probar imprimir el reporte. Probar la visualización del reporte desde varios navegadores.

HISTORIA DE USUARIO 43	
Número: H43	Nombre de la historia: El sistema permitirá generar un reporte de las asistencias por curso y materia.
Usuario: Inspector	Iteración Asignada: 6
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 17
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como inspector deseo poder generar un reporte de las asistencias por curso y materia.	
Observaciones: <ul style="list-style-type: none"> • El reporte será generado en formato PDF 	

Historia de Usuario 43 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de generación de reporte de las asistencias por curso y materia.	Probar la generación de reportes con la lista de estudiantes vacía. Probar que los datos calculados y reflejados en el reporte sean los correctos. Probar imprimir el reporte. Probar la visualización del reporte desde varios navegadores.

HISTORIA DE USUARIO 44	
Número: H44	Nombre de la historia: El sistema permitirá generar un reporte con el promedio general de materias, el promedio de exámenes de grado de cada estudiante.
Usuario: Inspector	Iteración Asignada: 6
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 18
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	
Descripción: Como secretaria deseo poder generar un reporte de que muestre el promedio general de materias, el promedio de exámenes de grado de cada estudiante.	
Observaciones: <ul style="list-style-type: none"> El reporte será generado en formato PDF 	

Historia de Usuario 44 - Pruebas de Aceptación	
Nombre de la prueba	Descripción
Test de generación de reporte de promedio general de materias y exámenes de grado.	Probar la generación de reportes con la lista de estudiantes vacía. Probar que los datos calculados y reflejados en el reporte sean los correctos. Probar imprimir el reporte. Probar la visualización del reporte desde varios navegadores.

- TAREAS DEL SPRINT 6**

TAREA DE INGENIERÍA	
Historia de Usuario: H40	
Número Tarea: 117	Nombre de la Tarea: Crear patrones de rutas para acceder al reporte de los estudiantes en suspenso.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Johnny Parra	
Descripción: <ul style="list-style-type: none"> Se creará la ruta en el archivo de enrutamiento que permita acceder al reporte de suspenso. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H40	
Número Tarea: 118	Nombre de la Tarea: Crear funciones y métodos para generar el reporte de los estudiantes en suspenso.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará métodos para consultar los estudiantes que están suspensos de un determinado curso y materia. • Se creará funciones que permita generar el reporte requerido 	

TAREA DE INGENIERÍA	
Historia de Usuario: H40	
Número Tarea: 119	Nombre de la Tarea: Crear la vista y/o diseño del reporte de notas de suspenso.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se diseñará el reporte conforme los requerimientos del sistema. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H41	
Número Tarea: 120	Nombre de la Tarea: Crear patrones de rutas para acceder al reporte de notas parciales
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará una ruta que permita acceder al reporte de notas parciales 	

TAREA DE INGENIERÍA	
Historia de Usuario: H41	
Número Tarea: 121	Nombre de la Tarea: Crear funciones y métodos para generar el reporte de notas parciales
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará métodos para consultar las notas parciales de un determinado, mes, materia y curso • Se creará funciones que permita generar el reporte requerido 	

TAREA DE INGENIERÍA	
Historia de Usuario: H41	
Número Tarea: 122	Nombre de la Tarea: Crear la vista y/o diseño del reporte de notas parciales
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se diseñará el reporte conforme los requerimientos del sistema 	

TAREA DE INGENIERÍA	
Historia de Usuario: H42	
Número Tarea: 123	Nombre de la Tarea: Crear patrones de rutas para acceder al reporte general de calificaciones de las notas parciales.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará la ruta en el archivo de enrutamiento que permita acceder al general de calificaciones de las notas parciales. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H42	
Número Tarea: 124	Nombre de la Tarea: Crear funciones y métodos para generar el reporte general de calificaciones de las notas parciales.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará métodos para consultar y calcular los promedios que tendrá el reporte, para un determinado curso y materia. • Se creará funciones que permita generar el reporte requerido. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H42	
Número Tarea: 125	Nombre de la Tarea: Crear la vista y/o diseño del reporte general de calificaciones de las notas parciales.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se diseñará el reporte conforme los requerimientos del sistema. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H43	
Número Tarea: 126	Nombre de la Tarea: Crear patrones de rutas para acceder al reporte de asistencia de estudiantes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción: Se creará una ruta que permita acceder al reporte de asistencia de estudiantes	

TAREA DE INGENIERÍA	
Historia de Usuario: H43	
Número Tarea: 127	Nombre de la Tarea: Crear funciones y métodos para generar el reporte de asistencias de estudiantes
Tipo de Tarea: Desarrollo	Puntos de Estimados: 8
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará métodos para consultar las asistencias de uno determinado curso, y materia. • Se creará funciones que permita generar el reporte requerido 	

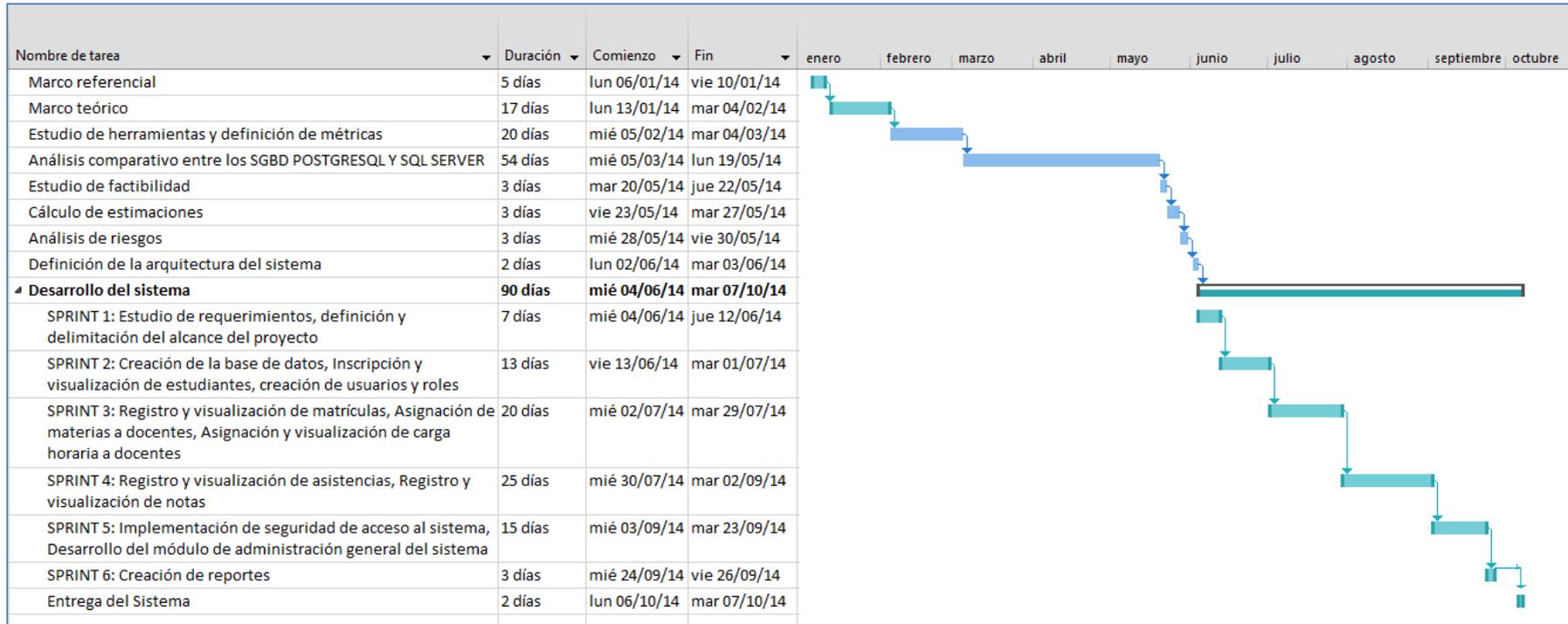
TAREA DE INGENIERÍA	
Historia de Usuario: H43	
Número Tarea: 128	Nombre de la Tarea: Crear la vista y/o diseño del reporte de asistencias
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se diseñará el reporte conforme los requerimientos del sistema 	

TAREA DE INGENIERÍA	
Historia de Usuario: H44	
Número Tarea: 129	Nombre de la Tarea: Crear patrones de rutas para acceder al reporte de promedios generales de materias y exámenes de grado.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 3
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se creará una ruta que permita acceder al reporte de promedios generales de materias y exámenes de grado. 	

TAREA DE INGENIERÍA	
Historia de Usuario: H44	
Número Tarea: 130	Nombre de la Tarea: Crear funciones y métodos para generar el reporte de promedios generales de materias y exámenes de grado.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 9
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Johnny Parra	
Descripción:	
<ul style="list-style-type: none"> • Se creará métodos para consultar los promedios generales de materias y exámenes de grado. • Se creará funciones que permita generar el reporte requerido 	

TAREA DE INGENIERÍA	
Historia de Usuario: H44	
Número Tarea: 131	Nombre de la Tarea: Crear la vista y/o diseño del reporte de promedios generales de materias y exámenes de grado.
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 24/09/2014	Fecha de Fin: 26/09/2014
Programador Responsable: Mario Andrade	
Descripción:	
<ul style="list-style-type: none"> • Se diseñará el reporte conforme los requerimientos del sistema 	

PLANIFICACIÓN GENERAL DEL SISTEMA



DICCIONARIO DE DATOS

Tabla	Columna	Tipo de dato	Tamaño	Nullable	Clave Primaria	Clave Foránea
asistencia	id	Int	4	No	Si	No
asistencia	materiaasignada_id	Int	4	Si	No	Si
asistencia	horasmodulo	Int	4	Si	No	No
asistencia	faltasjustificadas	Int	4	Si	No	No
asistencia	faltasinjustificadas	Int	4	Si	No	No
asistencia	atrasos	Int	4	Si	No	No
asistencia	horasasistidas	Int	4	Si	No	No
asistencia	observaciones	NVarChar (255)	255	Si	No	No
asistencia	promediofinal	Float	8	Si	No	No
cumplerequisito	id	Int	4	No	Si	No
cumplerequisito	requisito_id	Int	4	Si	No	Si
cumplerequisito	inscripcion_id	Int	4	Si	No	Si
cumplerequisito	estado	Bit	1	Si	No	No
Curso	id	Int	4	No	Si	No
Curso	nombre	NVarChar (32)	32	No	No	No
Dia	id	Int	4	No	Si	No
Dia	nombre	NVarChar (32)	32	No	No	No
dictadomateria	id	Int	4	No	Si	No
dictadomateria	nivel_id	Int	4	Si	No	Si
dictadomateria	docente_id	Int	4	Si	No	Si
dictadomateria	periodo_id	Int	4	Si	No	Si
dictadomateria	materiaperiodo_id	Int	4	Si	No	Si

docente	id	Int	4	No	Si	No
docente	cedula	Int	4	No	No	No
docente	nombre	NVarChar (64)	64	No	No	No
docente	apellido	NVarChar (64)	64	No	No	No
docente	direccion	NVarChar (128)	128	No	No	No
docente	telefonofijo	NVarChar (13)	13	Si	No	No
docente	celular	NVarChar (13)	13	Si	No	No
docente	email	NVarChar (128)	128	No	No	No
docente	trabajoanterior	NVarChar (128)	128	Si	No	No
docente	fechaingreso	Date	3	No	No	No
docente	doc_fechasalida	Date	3	Si	No	No
docente	estado	Bit	1	Si	No	No
docente	salt	NVarChar (255)	255	Si	No	No
docente	password	NVarChar (255)	255	Si	No	No
estudiante	id	Int	4	No	Si	No
estudiante	cedula	Int	4	No	No	No
estudiante	nombre	NVarChar (64)	64	No	No	No
estudiante	apellido	NVarChar (64)	64	No	No	No
estudiante	telefonofijo	NVarChar (13)	13	Si	No	No
estudiante	celular	NVarChar (13)	13	Si	No	No
estudiante	calle	NVarChar (128)	128	Si	No	No
estudiante	barrio	NVarChar (64)	64	Si	No	No
estudiante	parroquia	NVarChar (64)	64	Si	No	No
estudiante	ciudad	NVarChar (32)	32	Si	No	No
estudiante	email	NVarChar (128)	128	No	No	No
estudiante	ocupacion	NVarChar (64)	64	Si	No	No
estudiante	lugarnacimiento	NVarChar (32)	32	Si	No	No
estudiante	estado	NVarChar (32)	32	No	No	No
estudiante	path	NVarChar (255)	255	Si	No	No

evaluacion	id	Int	4	No	Si	No
evaluacion	materiaasignada_id	Int	4	Si	No	Si
evaluacion	mesevaluacion_id	Int	4	Si	No	Si
evaluacion	promedio	Float	8	No	No	No
evaluacion	descripcion	NVarChar (128)	128	Si	No	No
evaluacion	notatb	Float	8	No	No	No
evaluacion	notaec	Float	8	No	No	No
evaluacion	notapp	Float	8	No	No	No
evaluacion	notapt	Float	8	No	No	No
examengrado	id	Int	4	No	Si	No
examengrado	matricula_id	Int	4	Si	No	Si
examengrado	materiagrado_id	Int	4	Si	No	Si
examengrado	equivalencia	NVarChar (32)	32	Si	No	No
examengrado	descripcion	NVarChar (96)	96	Si	No	No
examengrado	nota	Float	8	No	No	No
hora	id	Int	4	No	Si	No
hora	hora_inicio	Time	3	No	No	No
hora	hora_fin	Time	3	No	No	No
hora	descripcion	NVarChar (64)	64	Si	No	No
horarioclase	id	Int	4	No	Si	No
horarioclase	dia_id	Int	4	Si	No	Si
horarioclase	hora_id	Int	4	Si	No	Si
horarioclase	dictadomateria_id	Int	4	Si	No	Si
horarioclase	observacion	NVarChar (64)	64	Si	No	No
inscripcion	id	Int	4	No	Si	No

inscripcion	estudiante_id	Int	4	Si	No	Si
inscripcion	periodo_id	Int	4	Si	No	Si
inscripcion	estado	Int	4	No	No	No
materia	id	Int	4	No	Si	No
materia	numerohoras	Int	4	Si	No	No
materia	nombre	NVarChar (128)	128	No	No	No
materia	numerocreditos	Int	4	Si	No	No
materia	descripcion	NVarChar (128)	128	Si	No	No
materia	estado	Bit	1	Si	No	No
materiaasignada	id	Int	4	No	Si	No
materiaasignada	matricula_id	Int	4	Si	No	Si
materiaasignada	materiaperiodo_id	Int	4	Si	No	Si
materiaasignada	equivalencia	NVarChar (32)	32	Si	No	No
materiaasignada	notasuspenseo	Float	8	Si	No	No
materiaasignada	promediofinal	Float	8	Si	No	No
materiagrado	id	Int	4	No	Si	No
materiagrado	materia_id	Int	4	Si	No	Si
materiagrado	periodo_id	Int	4	Si	No	Si
materiagrado	estado	Bit	1	Si	No	No
materiaperiodo	id	Int	4	No	Si	No
materiaperiodo	materia_id	Int	4	Si	No	Si
materiaperiodo	subperiodo_id	Int	4	Si	No	Si
matricula	Id	Int	4	No	Si	No
matricula	nivel_id	Int	4	Si	No	Si
matricula	periodo_id	Int	4	Si	No	Si
matricula	estudiante_id	Int	4	Si	No	Si
matricula	seccion	NVarChar (32)	32	No	No	No

matricula	fechamatricula	Date	3	No	No	No
matricula	estado	Int	4	No	No	No
matricula	observaciones	NVarChar (128)	128	Si	No	No
mesevaluacion	id	Int	4	No	Si	No
mesevaluacion	nota_id	Int	4	Si	No	Si
mesevaluacion	periodo_id	Int	4	Si	No	Si
mesevaluacion	finiciomes	Date	3	No	No	No
mesevaluacion	ffinmes	Date	3	No	No	No
mesevaluacion	estado	Bit	1	Si	No	No
nivel	id	Int	4	No	Si	No
nivel	curso_id	Int	4	Si	No	Si
nivel	paralelo_id	Int	4	Si	No	Si
nivel	descripcion	NVarChar (128)	128	Si	No	No
nota	id	Int	4	No	Si	No
nota	nombre	NVarChar (64)	64	No	No	No
nota	estado	Bit	1	Si	No	No
paralelo	id	Int	4	No	Si	No
paralelo	nombre	NVarChar (32)	32	No	No	No
periodo	id	Int	4	No	Si	No
periodo	nombre	NVarChar (64)	64	No	No	No
periodo	finicioperiodo	Date	3	No	No	No
periodo	ffinperiodo	Date	3	No	No	No
periodo	finiciomatricula	Date	3	Si	No	No
periodo	ffinmatricula	Date	3	Si	No	No
periodo	finicioinscripcion	Date	3	Si	No	No
periodo	ffininscripcion	Date	3	Si	No	No

periodo	estado	Bit	1	Si	No	No
promedio	id	Int	4	No	Si	No
promedio	matricula_id	Int	4	Si	No	Si
promedio	pmodulos	Float	8	No	No	No
promedio	pgrado	Float	8	No	No	No
promedio	pgeneral	Float	8	No	No	No
promedio	equivalencia	NVarChar (32)	32	Si	No	No
requisito	id	Int	4	No	Si	No
requisito	descripcion	NVarChar (128)	128	No	No	No
requisito	estado	Bit	1	Si	No	No
rol	id	Int	4	No	Si	No
rol	nombre	NVarChar (64)	64	No	No	No
rol	descripcion	NVarChar (128)	128	Si	No	No
subperiodo	id	Int	4	No	Si	No
subperiodo	periodo_id	Int	4	Si	No	Si
subperiodo	tipo	NVarChar (32)	32	No	No	No
subperiodo	finicio	Date	3	Si	No	No
subperiodo	ffin	Date	3	Si	No	No
subperiodo	estado	Bit	1	Si	No	No
usuario	id	Int	4	No	Si	No
usuario	rol_id	Int	4	Si	No	Si
usuario	nombre	NVarChar (128)	128	No	No	No
usuario	apellidos	NVarChar (128)	128	No	No	No

usuario	email	NVarChar (255)	255	No	No	No
usuario	password	NVarChar (255)	255	No	No	No
usuario	salt	NVarChar (255)	255	No	No	No
usuario	fecha_alta	DateTime2	8	No	No	No
usuario	cedula	Int	4	No	No	No
usuario	estado	Bit	1	No	No	No

BIBLIOGRAFÍA

- [1] ALEGSA. (2014). *alegsa.com.ar*. Obtenido de <http://www.alegsa.com.ar/Dic/rendimiento.php>
- [2] ANGARITA, F. (04 de 2014). *Microsoft Developer Network*. Obtenido de <http://msdn.microsoft.com/es-es/library/dn391746.aspx>
- [3] BLANCO MUÑOZ, E., & DORADO CERÓN, J. (16 de 11 de 2010). *WikiCE*. Obtenido de <http://osl2.uca.es/wikiCE/index.php/Scrum>
- [4] BURLESON CONSULTING . (04 de 2014). *BC Burleson Consulting*. Obtenido de http://www.dba-oracle.com/t_benchmark_testing.htm
- [5] CARDETT ASSOCIATES. (04 de 2014). *Advanced Query Tool*. Obtenido de <http://www.querytool.com/>
- [6] CORTÉS CORTÉS, W., MUÑOZ SALAZAR, J., & QUIROZ ARMIJO, M. (08 de 2011). *Universidad de los Lagos Campus Santiago*. Obtenido de <http://ulagos.files.wordpress.com/2011/08/informe-de-base-de-datos-evaluacion-del-desempeño-de-una-base-de-datos.pdf>
- [7] DATAPRIX. (2013). *DataPrix Knowledge Is The Goal*. Recuperado el 20 de Septiembre de 2013, de <http://www.dataprix.com/bases-datos-postgresql>
- [8] DATAPRIX. (2013). *DataPrix Knowledge Is The Goal*. Obtenido de <http://www.dataprix.com/en/taxonomy/term/740/all/feed>
- [9] DTM SOFT. (04 de 2014). *DTM Database Tools*. Obtenido de <http://www.sqledit.com/stress/>
- [10] DUBRETIC, M. (26 de 04 de 2014). *Udemy*. Obtenido de <https://www.udemy.com/blog/es/oracle-vs-mysql-vs-sql-server-una-comparacion-entre-los-sistemas-gestores-de-bases-de-datos-relacionales-mas-populares/>
- [11] Eguiluz, J. (2013). *Desarrollo web ágil con Symfony2*. España: Gestor de publicaciones EasyBook.
- [12] FRITCHEY, G. (31 de 05 de 2007). *Simple-Talk*. Obtenido de <https://www.simple-talk.com/sql/performance/sql-server-performance-crib-sheet/>
- [13] GUTIÉRREZ GALLARDO, J. D. (2011). *Desarrollo web con PHP 6 y MySQL 5.2*. Anaya Multimedia.
- [14] HAMMERDB. (06 de 2013). *HammerDB*. Recuperado el 2 de 10 de 2013, de <http://hammerora.sourceforge.net/document.html>
- [15] HURTADO, R., VILUÑIR, A., & TAPIA, E. (08 de 2011). *Universidad de los Lagos campus Santiago*. Obtenido de <http://ulagos.files.wordpress.com/2011/08/flexibilidad-y-portabilidad.doc>
- [16] IREO. (2013). *IREO*. Obtenido de http://www.ireo.com/fileadmin/docs/documentacion_de_productos/manageengine/applicationsmanager/ME_Applications%20Manager.pdf

- [17] ITEA SOLUCIONES TIC. (03 de 2014). *Itea Soluciones TIC*. Obtenido de http://www.cpraviles.com/materiales/pcpi/PCPI/indexfee9.html?page_id=891
- [18] LAMBERT SARANGO, Y. (31 de 05 de 2011). *YAMIL L^aM^a3ЯT - ESPOL - YAMIL LAMBERT*. Obtenido de <http://blog.espol.edu.ec/ylambert/2011/05/31/>
- [19] MACHANIC, A. (04 de 2014). *SQLblog.com*. Obtenido de <http://www.datamanipulation.net/sqlquerystress/>
- [20] MANENE, L. (15 de 04 de 2011). *Luis Miguel Manene*. Obtenido de <http://www.luismiguelmanene.com/2011/04/15/benchmarkingdefiniciones-aplicaciones-tipos-y-fases-del-proceso/>
- [21] MARTINEZ, R. (2 de 10 de 2010). *PostgreSQL-es*. Obtenido de http://www.postgresql.org.es/sobre_postgresql
- [22] MICROSOFT. (2012). *Microsoft Developer Network*. Recuperado el 2013 de 09 de 2013, de [http://msdn.microsoft.com/en-us/library/cc768530\(v=bts.10\).aspx](http://msdn.microsoft.com/en-us/library/cc768530(v=bts.10).aspx)
- [23] MICROSOFT. (2012). *Microsoft Developer Network*. Obtenido de <http://msdn.microsoft.com/es-es/library/ms345332.aspx>
- [24] MICROSOFT. (2012). *Microsoft Developer Network*. Obtenido de <http://msdn.microsoft.com/es-es/library/ms345343.aspx>
- [25] MICROSOFT. (2012). *Microsoft TechNet*. Obtenido de <http://technet.microsoft.com/es-es/library/ms184276%28v=sql.105%29.aspx>
- [26] MICROSOFT. (2012). *Microsoft TechNet*. Obtenido de [http://technet.microsoft.com/es-es/library/ms186312\(v=sql.110\).aspx](http://technet.microsoft.com/es-es/library/ms186312(v=sql.110).aspx)
- [27] MICROSOFT. (2013). *Microsoft SQL Server 2012*. Recuperado el 6 de 10 de 2013, de <http://technet.microsoft.com/en-US/sqlserver/ff898410>
- [28] MICROSOFT. (2013). *Microsoft TechNet*. Recuperado el 13 de 10 de 2013, de [http://technet.microsoft.com/es-es/library/ms190619\(v=sql.105\).aspx](http://technet.microsoft.com/es-es/library/ms190619(v=sql.105).aspx)
- [29] MICROSOFT. (2014). *Microsoft Developer Network*. Obtenido de <http://msdn.microsoft.com/es-es/library/ms174173.aspx>
- [30] MICROSOFT. (2014). *Microsoft Developer Network*. Obtenido de <http://msdn.microsoft.com/es-es/library/ms174212.aspx>
- [31] MICROSOFT. (2014). *Microsoft Developer Network*. Obtenido de <http://msdn.microsoft.com/es-es/library/ms162773.aspx>
- [32] MICROSOFT. (2014). *Microsoft TechNet*. Obtenido de [http://technet.microsoft.com/es-es/library/ms162169\(v=sql.110\).aspx](http://technet.microsoft.com/es-es/library/ms162169(v=sql.110).aspx)
- [33] MICROSOFT. (2014). *Microsoft TechNet*. Obtenido de <http://technet.microsoft.com/es-es/library/ms190623%28v=sql.105%29.aspx>
- [34] OBERTO, P. (24 de 03 de 2012). *Modelos de BD*. Obtenido de <https://modelosbd2012t1.wordpress.com/autor/pedrooberto/>

- [35] PÉREZ MARQUÉS, M. (2012). *SQL Server 2008 R2: Motor de base de datos y administración*. Alfaomega.
- [36] PILCO GUACHI, L. (6 de 2 de 2012). *Dspace*. Obtenido de <http://dspace.esPOCH.edu.ec/bitstream/123456789/1534/1/18T00475.pdf>
- [37] PINAL, D. (30 de 08 de 2012). *Journey to SQL Authority with Pinal Dave*. Obtenido de <http://blog.sqlauthority.com/2012/08/30/sql-server-beginning-of-sql-server-architecture-terminology-guest-post/>
- [38] PUENTE LANZAROTE, L., DEL CAMPO HILARIO, C., & RUIZ DE LUZURIAGA PEÑA, M. (s.f.). *e-prints in library & information science*. Recuperado el 12 de 10 de 2013, de <http://eprints.rclis.org/14991/1/Puente-Indicadores.pdf>
- [39] REINOZA, E. (2012). *Base de datos*. Alfaomega.
- [40] RODRÍGUEZ FLORES, I. (2011). *Texto básico: Base de datos I*. s.ed.
- [41] ROJAS MIRANDA, R. (29 de 10 de 2012). *Prezi*. Obtenido de <http://prezi.com/r1x0xccscnw/rendimiento-del-computador-parametros-principalesmedidas-del-rendimiento/>
- [42] ROSS, M., & MISNER, S. (2012). *Introducing Microsoft® SQL Server 2012; Ross Mistry and Stacia Misner*. Redmond, Washington 98052-6399: Microsoft Press.
- [43] SABANA MENDOZA, M. (2006). *PHP con PostgreSQL 8*. Megabyte.
- [44] THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. (18 de 05 de 2012). *PostgreSQL Wiki*. Obtenido de https://wiki.postgresql.org/wiki/Gu%C3%ADa_de_la_Comunidad_para_las_herramientas_GUI_de_PostgreSQL
- [45] THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. (2013). *PostgreSQL*. Obtenido de <http://www.postgresql.org/about/>
- [46] THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. (2013). *PostgreSQL*. Obtenido de <http://www.postgresql.org/about/press/presskit92/es/>
- [47] ZOHOO CORPORATION. (2013). *PostgreSQL Monitoring*. Recuperado el 4 de 10 de 2013, de http://www.manageengine.com/products/applications_manager/postgresql-monitoring.html