



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

**“DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN Y DIFUSIÓN
DE LOS EMPRENDIMIENTOS EN LA FUNDACIÓN “EMPRENDER Y NO
DEPENDER””**

TRABAJO DE TITULACIÓN
TIPO: PROYECTO TÉCNICO

Presentado para optar al grado académico de
INGENIERA EN SISTEMAS INFORMÁTICOS

AUTORA: ADRIANA CECIBEL QUILLE PUNINA
TUTOR: ING. EDUARDO ROLANDO VILLA VILLA

Riobamba – Ecuador
2019

© 2019, Adriana Cecibel Quille Punina

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

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

El tribunal de Trabajo de Titulación certifica que: El proyecto Técnico “DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN Y DIFUSIÓN DE LOS EMPRENDIMIENTOS EN LA FUNDACIÓN “EMPRENDER Y NO DEPENDER””, de responsabilidad de la señorita ADRIANA CECIBEL QUILLE PUNINA, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, quedando autorizada su presentación.

NOMBRE

FIRMA

FECHA

Ing. Washington Luna

**DECANO DE LA FACULTAD DE
INFORMÁTICA Y ELECTRÓNICA**

Ing. Patricio Moreno

**DIRECTOR DE LA ESCUELA
DE INGENIERÍA EN SISTEMAS**

Ing. Eduardo Villa

**DIRECTOR DE TRABAJO
DE TITULACIÓN**

Ing. Alejandra Oñate

**MIEMBRO DE TRABAJO
DE TITULACIÓN**

Yo, ADRIANA CECIBEL QUILLE PUNINA soy responsable de las ideas, doctrinas y resultados expuestos en esta Tesis y el patrimonio intelectual de la Tesis de Grado pertenece a la Escuela Superior Politécnica de Chimborazo.

Adriana Cecibel Quille Punina

DEDICATORIA

Dedico este trabajo de titulación en primer lugar a Dios por brindarme salud y vida para poder realizar mis actividades sin ningún inconveniente, a mis padres Luis Ángel Quille y María Elvia Punina por apoyarme siempre en todos los objetivos que me propongo realizar y a su vez ser mis pilares fundamentales, para seguir adelante, gracias por sus enseñanzas, consejos que me han inculcado desde pequeña, a mis hermanos Odalis, Mayra, Gustavo que son mi inspiración y por haberme dado palabras alentadoras las cuales llevaré en mi mente y en mi corazón, a mi sobrina Kamila por sacarme un sonrisa en los momentos más difíciles por ser ese pequeño mundo que me acompaña siempre, a mis amigas Ely, Marthita, Majitho por demostrarme una amistad sincera y duradera y a esa persona especial I.C a quien lo llevo siempre conmigo, aquel que me apoyo moralmente para seguir en este caminar y a ver la vida desde otras perspectivas y a todos ustedes les dedico este esfuerzo para la culminación de mi carrera profesional.

Adriana.

AGRADECIMIENTO

Agradezco a Dios por haberme regalado la vida junto a ello nuevas oportunidades que me brinda día a día para seguir adelante en este caminar junto con la bendición de mis padres quienes han sido los entes principales impulsándome en mi preparación y toma de decisiones, agradecer a mi querida institución a la Escuela Superior Politécnica de Chimborazo por acogerme en este caminar propuesto a sus profesionales que han sido parte de mis aprendizajes para mi profesión los mismos que me han formado no solo como profesional sino también como una persona para afrontar retos que la vida presenta, en especial al Ing. Eduardo Villa, por guiarme durante el proceso de desarrollo de este trabajo con sus consejos, anécdotas que jamás olvidaré, a la Ing. Alejandra Oñate por sus conocimientos compartidos para poder sacar un buen trabajo, sus recomendaciones y además inculcarme el deber de un profesional.

Con todo el corazón les agradezco infinitamente.

Adriana Quille

TABLA DE CONTENIDO

ÍNDICE DE TABLAS	xi
ÍNDICE DE FIGURAS	xiii
ÍNDICE DE GRÁFICOS	xv
ÍNDICE DE ANEXOS	xvi
ÍNDICE DE ABREVIATURAS	xvii
RESUMEN	xviii
ABSTRACT	xix
INTRODUCCIÓN	1

CAPÍTULO I

1. MARCO TEÓRICO	7
1.1. Sistema web	7
<i>1.1.1. Definición</i>	<i>7</i>
<i>1.1.2. Ventajas y desventajas de los sistemas web</i>	<i>7</i>
1.2. Sitio web orientado a los Emprendimientos	8
1.3. Arquitecturas de software	9
<i>1.3.1. Características</i>	<i>9</i>
<i>1.3.2. Comparativa de las Arquitecturas de software</i>	<i>10</i>
1.4. Arquitectura cliente/servidor	11
<i>1.4.1. Definición</i>	<i>11</i>
<i>1.4.2. Características</i>	<i>12</i>
<i>1.4.3. Ventajas y desventajas</i>	<i>12</i>
1.5. Patrón de Diseño MVC (Modelo Vista Controlador)	13
<i>1.5.1. Definición</i>	<i>13</i>
<i>1.5.2. Elementos MVC</i>	<i>13</i>
<i>1.5.3. Labor del patrón MVC.</i>	<i>14</i>
<i>1.5.4. Ventajas y desventajas del patrón MVC</i>	<i>15</i>
1.6. Lenguajes de Programación orientados a la web	15
<i>1.6.1. Comparativa entre diferentes lenguajes de programación orientados a la web</i>	<i>16</i>
1.7. PHP	17
<i>1.7.1. Definición</i>	<i>17</i>
<i>1.7.2. Características</i>	<i>17</i>

1.7.3.	<i>Ventajas</i>	18
1.8.	Framework de Desarrollo	19
1.8.1.	<i>Definición</i>	19
1.8.2.	<i>Ventajas y Desventajas</i>	19
1.8.3.	<i>Tipos de Framework para PHP</i>	20
1.8.4.	<i>Comparativa de Framework para PHP</i>	21
1.9.	Symfony	22
1.9.1.	<i>Características</i>	22
1.9.2.	<i>Estructura de directorios en Symfony</i>	23
1.9.3.	<i>Implementación de MVC que realiza Symfony</i>	24
1.10.	Bootstrap	24
1.10.1.	<i>Definición</i>	24
1.10.2.	<i>Ventajas y desventajas de Bootstrap</i>	25
1.11.	Tecnología para el Desarrollo de Mensajería	26
1.11.1.	<i>Definición de mensajería instantánea</i>	26
1.11.2.	<i>Web Socket</i>	26
1.11.3.	<i>Ventajas y Desventajas de WebSocket</i>	26
1.11.4.	<i>Comparativa de las tecnologías para el chat en línea.</i>	27
1.12.	Base de Datos	28
1.12.1.	<i>Definición</i>	28
1.12.2.	<i>Comparativa entre bases de datos</i>	29
1.12.3.	<i>MySQL</i>	30
1.12.4.	<i>Características</i>	30
1.12.5.	<i>Ventajas y desventajas</i>	31
1.13.	Servidor Apache	31
1.13.1.	<i>Características</i>	32
1.13.2.	<i>Ventajas y desventajas del servidor Apache</i>	32
1.14.	Metodología de desarrollo	32
1.14.1.	<i>Características</i>	33
1.14.2.	<i>Comparativa de metodologías tradicional vs ágil</i>	33
1.15.	Metodologías Ágiles	34
1.15.1.	<i>Características</i>	34
1.15.2.	<i>Comparativa de metodologías ágiles</i>	34
1.16.	SCRUM	35

1.16.1.	<i>Características</i>	35
1.16.2.	<i>Roles de SCRUM</i>	36
1.17.	Norma ISO/IEC 9126	38
1.17.1.	<i>Características de la ISO/IEC 9126</i>	39
1.17.2.	<i>Eficiencia</i>	41
CAPÍTULO II		
2.	MARCO METODOLÓGICO	43
2.1.	Diseño de la Investigación	43
2.1.1.	<i>Tipo de investigación</i>	43
2.1.2.	<i>Métodos de investigación</i>	43
2.1.3.	<i>Técnicas de investigación</i>	44
2.2.	Determinación de los procesos en la Fundación “Emprender y No Dependier” ...	44
2.2.1.	<i>Diagrama de procesos</i>	44
2.2.2.	<i>Diagrama de procesos sistematizado</i>	48
2.3.	Desarrollo de la metodología ágil SCRUM	50
2.3.1.	<i>Fase de planificación</i>	50
2.3.1.1.	<i>Personas involucradas</i>	50
2.3.1.2.	<i>Tipos y roles de usuario</i>	50
2.3.1.3.	<i>Planificación</i>	51
2.3.1.4.	<i>Producto Backlog</i>	51
2.3.1.5.	<i>Sprint backlog</i>	54
2.3.1.6.	<i>Reuniones SCRUM</i>	56
2.3.2.	<i>Fase de desarrollo</i>	57
2.3.2.1.	<i>Gestión de riesgos</i>	57
2.3.2.2.	<i>Arquitectura del sistema</i>	58
2.3.2.3.	<i>Diseño del sistema web</i>	59
2.3.2.4.	<i>Estándar de codificación</i>	60
2.3.2.5.	<i>Estándar de Interfaces</i>	60
2.3.2.6.	<i>Diseño de la Base de datos</i>	62
2.3.2.7.	<i>Diccionario de datos</i>	65
2.3.2.8.	<i>Instalación, configuración y codificación</i>	65
2.3.2.9.	<i>Historias de usuario</i>	74
2.3.2.10.	<i>Manual de usuario</i>	76
2.3.3.	<i>Fase de Finalización</i>	77

2.3.3.1.	<i>Burndown Chart</i>	77
2.4.	Métodos utilizados para evaluar la eficiencia del sistema web	78
2.4.1.	<i>Análisis de la eficiencia del sistema</i>	78
2.4.2.	<i>Población</i>	78
2.4.3.	<i>Muestra</i>	78
2.4.4.	<i>Comportamiento de tiempos</i>	79
2.4.5.	<i>Obtención de datos</i>	79

CAPÍTULO III

3.	MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS	81
3.1.1.	<i>Análisis de la normalidad de los datos</i>	81
3.1.2.	<i>Caso 1</i>	85
3.1.2.1.	<i>Estadística Descriptiva</i>	85
3.1.2.2.	<i>Estadística Inferencial</i>	86
3.1.3.	<i>Caso 2</i>	88
3.1.3.1.	<i>Estadística Descriptiva</i>	89
3.1.3.2.	<i>Estadística Inferencial</i>	89
3.1.3.3.	<i>Estadística Descriptiva</i>	92
3.1.3.4.	<i>Estadística Inferencial</i>	93

CONCLUSIONES	96
---------------------------	----

RECOMENDACIONES	97
------------------------------	----

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 1-1:	Ventajas y desventajas de los sistemas web.	8
Tabla 2-1:	Comparativa de arquitecturas	10
Tabla 3-1:	Ventajas y desventajas de la arquitectura cliente/servidor.	12
Tabla 4-1:	Ventajas y desventajas de MVC.	15
Tabla 5-1:	Comparativa de lenguajes de programación.	16
Tabla 6-1:	Ventajas de PHP.....	18
Tabla 7-1:	Ventajas y desventajas de los framework.....	19
Tabla 8-1:	Comparativa de Framework.....	21
Tabla 9-1:	Ventajas y desventajas de Bootstrap.	25
Tabla 10-1:	Tecnologías para chat en línea.	28
Tabla 11-1:	Comparativa de Bases de Datos.....	29
Tabla 12-1:	Ventajas y desventajas de MySQL.....	31
Tabla 13-1:	Ventajas y desventajas del servidor Apache.....	32
Tabla 14-1:	Comparativa de metodologías tradicional vs ágil	33
Tabla 15-1:	Comparativa de metodologías ágiles.....	34
Tabla 1-2:	Procesos de la Fundación de manera no automatizada.....	44
Tabla 3-2:	Roles y personas	50
Tabla 4-2:	Tipos de usuario y roles.....	50
Tabla 5-2:	Estimaciones para el sistema.	51
Tabla 6-2:	Producto backlog del sistema web.	52
Tabla 7-2:	Sprint backlog del sistema web.....	54
Tabla 8-2:	Riesgos identificados en el proyecto.....	57
Tabla 9-2:	Estándar de codificación.....	60
Tabla 10-2:	Tabla cliente.....	65
Tabla 11-2:	Historia de usuario 01: Registrar datos de los productos.....	75
Tabla 12-2:	Prueba de aceptación de la HU_01.	76
Tabla 13-2:	Tarea de ingeniería 01 de la HU_01.....	76
Tabla 14-2:	Actividades de finalización del proyecto.....	77
Tabla 1-3:	Resultados de tiempos en segundos para la toma de decisión	85
Tabla 2-3:	Estadística descriptiva del registro de pedidos	86
Tabla 3-3:	Prueba T del registro de pedidos.....	87
Tabla 4-3:	Regla de decisión del registro de pedidos.....	87
Tabla 5-3:	Estadística descriptiva del registro de datos de los emprendimientos	89
Tabla 6-3:	Prueba T del registro de datos de los emprendimientos	90

Tabla 7-3:	Regla de decisión del registro de datos de los emprendimientos	91
Tabla 8-3:	Estadística descriptiva del registro de datos de los emprendedores	93
Tabla 9-3:	Prueba T del registro de datos de los emprendedores	94
Tabla 10-3:	Regla de decisión del registro de datos de los emprendedores	94

ÍNDICE DE FIGURAS

Figura 1-1:	Partes de un sistema web	7
Figura 2-1:	Arquitectura cliente – servidor	11
Figura 3-1:	Arquitectura MVC.....	14
Figura 4-1:	Arquitectura PHP.....	17
Figura 5-1:	Estructura de directorios de Symfony 4.....	23
Figura 6-1:	Arquitectura MVC en Symfony	24
Figura 7-1:	Roles de la metodología SCRUM.....	36
Figura 8-1:	Artefactos de la metodología SCRUM	37
Figura 9-1:	Características de la ISO/IEC 9126.....	39
Figura 10-1:	Resumen de la ISO/IEC 9126.....	41
Figura 1-2:	Diagrama de procesos parte 1.....	46
Figura 2-2:	Diagrama de procesos parte 2.....	47
Figura 3-2:	Diagrama de procesos sistematizado parte 1.....	48
Figura 4-2:	Diagrama de procesos sistematizado parte 2.....	49
Figura 5-2:	Arquitectura del sistema web.....	58
Figura 6-1:	Sistema Automatizado.....	59
Figura 7-2:	Página de Inicio.....	61
Figura 8-2:	Modal del ingreso de emprendimientos	61
Figura 9-2:	Interfaz del emprendimiento agregado.....	62
Figura 10-2:	Modelo lógico de la base de datos.....	64
Figura 11-2:	Descargar Composer para Windows.....	66
Figura 12-2:	Instalación de Composer.....	66
Figura 13-2:	Seleccionamos la carpeta donde se encuentra instalado el PHP.....	67
Figura 14-2:	Instalación de Composer	67
Figura 15-2:	Ingresando la línea de comandos.....	68
Figura 16-2:	Descarga todos los componentes que Symfony trae por defecto.....	68
Figura 17-2:	Instalación de todos los componentes descargados anteriormente.....	68
Figura 18-2:	Instalación del Framework en el directorio fundación.....	69
Figura 19-2:	Directorio del proyecto a desarrollar	69
Figura 20-2:	Página de bienvenida del framework Symfony.....	70
Figura 21-2:	Estructura de las carpetas del sistema.....	70
Figura 22-2:	Estructura MVC del sistema	71
Figura 23-2:	Controller de categorías del sistema.....	71
Figura 24-2:	Modelo de categoría del sistema	72

Figura 25-2: Formulario de la categoría del sistema.	72
Figura 26-2: Repository del sistema.	73
Figura 27-2: Habilitando la extensión socket.	74
Figura 28-2: Creación el servidor socket.	74
Figura 1-3: Tiempos tradicionales del registro de pedidos	82
Figura 2-3: Tiempos sistematizados del registro de pedidos.....	82
Figura 3-3: Tiempos tradicionales del registro de datos de los emprendimientos	83
Figura 4-3: Tiempos sistematizados del registro de datos de los emprendimientos	83
Figura 5-3: Tiempos tradicionales del registro de emprendedores	84
Figura 6-3: Tiempos sistematizados del registro de emprendedores	84

ÍNDICE DE GRÁFICOS

Gráfico 1-2:	Burndown Chart.....	77
Gráfico 1-3:	Gráfica de distribución T del registro de pedidos	87
Gráfico 2-3:	Tiempo promedio del registro de pedidos.....	88
Gráfico 3-3:	Gráfica de distribución T del registro de los emprendimientos	91
Gráfico 4-3:	Tiempos promedios del Registro de datos de los emprendimientos.....	92
Gráfico 5-3:	Gráfica de distribución T del registro del emprendedor.	94
Gráfico 6-3:	Tiempos promedios del registro de datos de los emprendedores	95

ÍNDICE DE ANEXOS

- ANEXO A:** Informe sobre las reuniones del equipo de desarrollo - cliente
- ANEXO B:** Análisis de riesgos
- ANEXO C:** Estándar de interfaces
- ANEXO D:** Diccionario de datos
- ANEXO E:** Manual técnico
- ANEXO F:** Manual de usuario
- ANEXO G:** Toma de tiempos para los resultados.

ÍNDICE DE ABREVIATURAS

SCRUM	Metodología ágil de desarrollo de software
NORMA ISO/IEC 9126	Normas para la evaluación de la eficiencia de software
MVC	Modelo vista controlador (Model View Controller)
PHP	Procesador de hipertexto (Hypertext Preprocessor)
SQL	Structured Query Language
CSS	Cascading style sheet – Hoja de estilo en cascada.
HTML	Hypertext markup Language – Lenguaje de marcas de hipertexto
HTTP	Hypertext Transfer Protocol – Protocolo de transferencia de hipertexto
BD	Base de datos
HU	Historia de Usuario
HT	Historia Técnica
TI	Tarea de Ingeniería
PA	Prueba de aceptación
IDE	Entorno de desarrollo integrado

RESUMEN

El presente trabajo de titulación tuvo como objetivo desarrollar un sistema web para la gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”, de la ciudad de Riobamba. Para la fase de requerimentación se aplicó la entrevista como técnica de investigación, la misma que facilitó recolectar la información de los procesos tradicionales que lleva la fundación, por otro lado mediante la observación se pudo analizar y recopilar los tiempos de respuesta, además se obtuvieron cuarenta y un (41) historias de usuarios, ocho (8) historias técnicas, las mismas que fueron planificadas y desarrolladas empleando la metodología ágil SCRUM, todas estas agrupadas en doce (12) Sprint con un total de novecientos sesenta (960) horas, ciento ochenta y dos (182) tareas de ingeniería y doscientas setenta y seis (276) pruebas de aceptación resultando todas estas como exitosas. También se empleó la Arquitectura Cliente/Servidor acompañada del patrón de diseño Modelo Vista Controlador (MVC), las herramientas tecnológicas para el desarrollo del sistema web fueron el framework Symfony basado en el lenguaje de programación PHP, un gestor de base de datos de MySQL, framework para maquetación Bootstrap, también para el desarrollo del chat en línea se utilizó WebSocket a través de la librería PHPWebSocket y finalmente para la evaluación de la eficiencia a través de la norma ISO/IEC 9126 se evaluó el comportamiento de los tiempos de respuesta basados en los procesos de registrar pedidos, registrar datos de los emprendimientos, y emprendedores respectivamente, obteniendo como resultado que la eficiencia en los procesos analizados ha mejorado en un promedio del 59,85% con un nivel de significancia de 0.05. Para una mejor difusión de emprendimientos se recomienda utilizar el chat en línea ya que este módulo permitirá interactuar con el administrador de la fundación.

PALABRAS CLAVES: <INGENIERÍA DE SOFTWARE>, <SISTEMA WEB>, <PROGRAMACIÓN WEB>, <GESTIÓN DE PRODUCTOS>, <NORMA ISO/IEC 9126>, <CHAT EN TIEMPO REAL>, <EFICIENCIA>, <METODOLOGÍA DE DESARROLLO ÁGIL (SCRUM)>.

ABSTRACT

This degree work aimed to develop a web system for the management and dissemination of the undertakings in "Emprender y No Dependier" Foundation, in Riobamba city. For the application phase, the interview was applied as a research technique, which facilitated the collection of information on the traditional processes carried out by the foundation. On the other hand, it was possible to analyze and collect response times through observation. In addition, forty-one (41) user stories were obtained, eight (8) technical histories, the same ones that were planned and developed using the agile SCRUM methodology, all grouped in twelve (12) Sprint with a total of nine hundred and sixty (960) hours, one hundred eighty-two (182) engineering tasks and two hundred and seventy-six (276) acceptance tests resulting as successful. The Client / Server Architecture was also used, accompanied by the design model Model Vista Controller (MVC). The technological tools for the development of the web system were the Symfony framework based on the PHP programming language, a MySQL database manager, framework for Bootstrap layout, also for the development of online chat, WebSocket was used through the PHPWebSocket library and finally for the evaluation of efficiency through ISO / IEC 9126 standard. The behavior of the response times was evaluated based on the processes of registering orders, recording data of the enterprises, and entrepreneurs respectively, obtaining as a result that the efficiency in the operations analyzed has improved by an average of 59.85% with a significance level of 0.05. For a better diffusion of undertakings, it is recommended to use online chat since this module will allow interaction with the administrator of the foundation.

Keywords: <SOFTWARE ENGINEERING>, <WEB SYSTEM>, <WEB PROGRAMMING>, <PRODUCT MANAGEMENT>, <ISO/IEC 9126 STANDARD>, <REAL-TIME CHAT>, <EFFICIENCY>, <AGIL DEVELOPMENT METHODOLOGY (SCRUM)>.

INTRODUCCIÓN

En la actualidad como se puede observar vivimos en un mundo que avanza a pasos agigantados no solo de manera social sino también de manera tecnológica la misma que ha venido dando pasos agigantados para obtener una mejor facilidad de comunicarse, organizarse, buscar y ofrecer los emprendimientos a todo el mundo, siendo la tecnología el medio principal para hacer cumplir esta meta.

Actualmente en las fundaciones es necesario contar con sistemas vigorosos que ayuden a optimizar los respectivos procesos satisfaciendo las necesidades del negocio mediante una automatización de información.

Uno de los propósitos principales que tienen las fundaciones es alcanzar un nivel de competitividad de gran escala en comparación con los demás, de manera que ayude a incentivar a más clientes y a crecer económicamente, para lo cual deben buscar las diversas maneras de dar a conocer cada uno de los clientes mediante campañas publicitarias que pueden ser distribuidas a través de diferentes medios como lo es en este caso la Tecnología.

Un sistema web facilitará los procesos que se lo realiza en las fundaciones mediante la gestión y difusión de sus emprendimientos, así como también centrarnos en la idea pertinente de la fundación para lograr reducir tiempos y ganar mayor productividad.

La Fundación “Emprender y No Dependier”, es una institución dedicada a la gestión y difusión de emprendimientos con pequeños microempresarios de la ciudad de Riobamba, todos estos procesos lo realizan a través de panfletos, llamadas telefónicas y visitas personalizadas, es decir de forma no automatizada, lo cual provoca que los emprendedores destinen demasiado tiempo en la promoción y venta de sus productos, por tanto se toma la decisión de desarrollar un sistema web para la gestión y difusión de los emprendimientos con la finalidad de reducir tiempos en la adquisición de los emprendimientos de esta fundación.

Para el desarrollo de este sistema web se observó algunos módulos que formaran parte de este, a continuación, se menciona:

El **módulo de usuarios** orientado específicamente al manejo de los usuarios que podrán acceder y manipular el sistema, el **módulo de emprendimientos** (producto) se lo realizo con el objetivo de llevar el respectivo control de datos de cada uno de los emprendimientos que posee la fundación, el **módulo de pedidos** se podrá realizar pedidos desde cualquier punto que se encuentre ubicado, en el **módulo de chat en línea** se lo realizo con la finalidad de mantener contacto con la fundación para obtener información más detallada acerca de un emprendimiento, el **módulo de calificación** del emprendimiento permitirá que el emprendedor conozca cuan acogida tiene su emprendimiento, en el **módulo de reportes** se podrá revisar la información más relevante de manera detallada de cada una de las actividades que se ha realizado, los **módulos de clientes y emprendedores** se encargarán de la **gestión** de datos tanto de clientes como emprendedores que pertenecen a la fundación.

ANTECEDENTES

Los emprendimientos sostenibles revisten especial importancia, de manera integral ayuda a paliar las deficiencias en el nivel de desarrollo de las comunidades.

Los emprendimientos en la Fundación “Emprender y No Dependier” tienen como prioridad dar a conocer a la ciudadanía y posteriormente al mundo entero, cada uno de estos productos a fin de lograr “unir, crecer e incluir” a empresarios y empresarias en una ruta que permita:

- Unir a la micro y pequeña empresa en unidades económicas asociadas, apoyarlas en sus esfuerzos de gremializarían y de encadenamientos productivos.
- Crecer es generar las oportunidades que permitan mejorar el entorno, la competitividad y la productividad de los emprendedores.
- Incluir a la micro y pequeña empresa en los procesos de desarrollo local, regional, nacional y en su inserción al mercado local, regional e internacional con la difusión de cada uno de los emprendimientos (Claros, 2017, p6).

Los framework web son un conjunto de herramientas, estilos y librerías dispuestas a través de una estructura o esqueleto base, para el desarrollo de aplicaciones web más escalables y sencillas de mantener (Studio, 2018, <https://www.illusionstudio.es/que-es-un-framework-web>).

Hasta hace pocos años, el uso de framework sólo era habitual para tareas de programación «pura» (por ejemplo, Symfony facilita la codificación en PHP). Sin embargo, actualmente existe un auge de los framework para maquetado y scripting, es decir, las áreas correspondientes al desarrollo front – end. Soluciones integrales como Bootstrap que brindan un generoso conjunto de elementos ya implementados en HTML, CSS y JavaScript para que el desarrollador pueda obtener sitios web atractivos en poco tiempo, Symfony diseñado para optimizar el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja (¿Es conveniente utilizar frameworks para desarrollo front-end?, 2013, <http://www.4rsoluciones.com/blog/es-conveniente-utilizar-frameworks-para-desarrollo-front-end-2/>).

Por lo tanto, en el presente proyecto se plantea la elaboración del Desarrollo de un sistema web para la gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”, para optimizar tiempos y lograr una administración y organización adecuada de los emprendimientos.

FORMULACIÓN DEL PROBLEMA

Actualmente en la Fundación “Emprender y No Dependier” se maneja la gestión y difusión de los productos creados por los emprendedores, a todos los potenciales clientes a través de panfletos, llamadas telefónicas y visitas personalizadas, es decir de forma no automatizada, lo cual provoca que los emprendedores destinen demasiado tiempo en la promoción y venta de sus productos, ocasionando problemas económicos y de gestión de los emprendimientos.

¿El sistema web para la Fundación “Emprender y No Dependier” optimizará los tiempos de gestión y difusión de los emprendimientos?

SISTEMATIZACIÓN DEL PROBLEMA

- ¿Cómo se lleva a cabo los procesos para la gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”?
- ¿Qué herramientas se podría utilizar para el desarrollo del sistema web?
- ¿Qué metodología, patrón de diseño y herramientas se podrán aplicar para el desarrollo del sistema web?
- ¿Cómo mejorará la eficiencia en el proceso de gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”?

JUSTIFICACIÓN DE LA INVESTIGACIÓN

JUSTIFICACIÓN TEÓRICA

Un sistema web le permitirá ahorrar costos, tiempo y recursos humanos, por tanto, la rentabilidad de la empresa aumentará y gestionará mejor a sus clientes, proveedores, distribuidores, etc. (Cañavate, 2014, pp.1-2).

Los sistemas web han demostrado mejores resultados para las empresas frente a los sistemas tradicionales Cliente/Servidor, pues los sistemas web brindan beneficios mientras que los tradicionales no pueden otorgar beneficios como:

- No pagar Licencias por cada computadora con el sistema instalado pues está en un sólo servidor web.
- Facilidad para acceder al sistema desde cualquier punto con conexión a internet
- Puede Integrar todas sus tiendas y sucursales; manteniendo información actualizada de stock de productos y de los ingresos y los egresos.
- Le permite comunicarse con sus clientes de manera más fluida las 24 horas del día (Peru, 2018, <https://websystemperu.com/sistemas-web>).

En la creación de un sistema web se trabaja con los elementos como un engranaje, en una página/aplicación web cohabitan lo que es el front – end y el back – end. El front – end y el back

– end lo que hacen es dividir la capa visual de un sistema y la capa del acceso de datos (Gómez, 2018, <https://steemit.com/spanish/@angelggomz/desarrollo-de-un-sistema-web-elementos-front-end-y-backend>).

Con la finalidad de dar solución al problema ya mencionado con anterioridad se desarrollará un sistema web para la gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier” haciendo uso del lenguaje de programación PHP, este es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor, para almacenar los datos y mantener su integridad, disponibilidad se utilizará MySQL el cual es un gestor de base de datos relacional desarrollado bajo licencia dual.

Para el diseño de las interfaces de usará el framework Bootstrap que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sistema web se adaptará automáticamente al tamaño de una pc, una tablet u otro dispositivo.

También para el lado del servidor se utilizará el framework Symfony el mismo que simplifica el desarrollo de las aplicaciones, automatiza muchos de los patrones utilizados para resolver las tareas comunes. Además, un framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y fácil de mantener. Por último, el framework facilita la programación de aplicaciones encapsulando operaciones complejas en instrucciones sencillas (Uniwebsidad, 2018, <https://uniwebsidad.com/libros/symfony-1-4/capitulo-1/symfony-en-pocas-palabras?from=librosweb>).

JUSTIFICACIÓN APLICATIVA

Es necesario el desarrollo de un sistema web para la gestión y difusión de los emprendimientos el mismo que permitirá que la micro y pequeña empresa pueda exhibir sus emprendimientos para obtener mejores beneficios y poder alcanzar en un futuro la internacionalidad de este.

Las funcionalidades básicas de este sistema web se basan en la gestión y difusión de los emprendimientos, la misma que posee alcanzar gran rentabilidad en el mercado para así promocionarles tanto nacional como internacional.

Los módulos para implementar son los siguientes:

- Módulo Autenticación de Usuarios

- Módulo de Gestión de Usuarios
- Módulo de Gestión de Productos
- Módulo de Chat en Línea para los emprendimientos
- Módulo de Calificación del Emprendedor

El presente software se desarrollará siguiendo las líneas de investigación número 5 de la ESPOCH de las tecnologías de la información, comunicaciones, procesos industriales y biotecnológicos dentro de la cual se encuentra el programa para el desarrollo de aplicaciones de software, hardware y telecomunicaciones.

Y bajo los parámetros del Programa Nacional Toda una vida.

OBJETIVOS

OBJETIVO GENERAL

Desarrollar un sistema web para la gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”, en la ciudad de Riobamba.

OBJETIVOS ESPECÍFICOS

- Analizar la situación actual de los procesos de gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”.
- Estudiar el framework Symfony y la tecnología WebSocket para el desarrollo ágil del sistema web.
- Aplicar la metodología ágil SCRUM, el patrón MVC, el framework Bootstrap, Symfony y WebSocket para el desarrollo del sistema web de gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”.
- Evaluar la eficiencia del sistema web en el proceso de gestión y difusión de los emprendimientos, utilizando la norma ISO 9126.

CAPÍTULO I

1. MARCO TEÓRICO

Al dar inicio al proceso de una investigación siempre se encontrará con términos poco habituales que no se utiliza a diario pero que son de gran importancia para que pueda ser utilizado en el área del saber en el que se investiga, a continuación, se presentará algunos selectos temas para el desarrollo de este trabajo de titulación.

1.1. Sistema web

1.1.1. Definición

Sistema web es una aplicación (Software) que no necesariamente se encuentra instalada en la computadora personal sino en otro equipo que hace las veces de servidor en la red **Figura 1-1**, este sistema web permite que los usuarios puedan acceder mediante un navegador web y mantener una comunicación activa entre el usuario y la información, por ejemplo, llenar y enviar formularios y, acceder a gestores de la base de datos. (Maldonado, 2016, p.12)

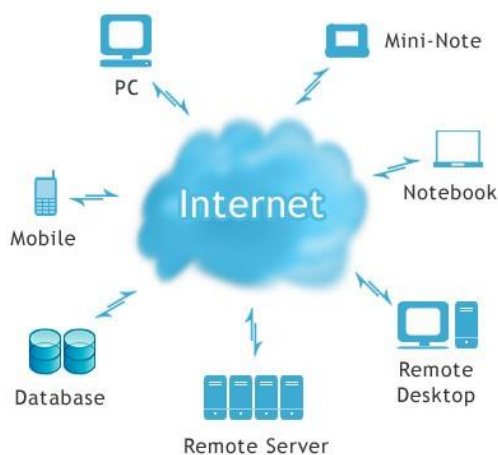


Figura 1-1: Partes de un sistema web

Fuente: (Baez, 2012, <http://www.knowdo.org/knowledge/39-sistemas-web>).

1.1.2. Ventajas y desventajas de los sistemas web

Un sistema web posee ventajas y desventajas que se presenta a continuación en la **Tabla 1-1**.

Tabla 1-1: Ventajas y desventajas de los sistemas web.

Ventajas	Desventajas
<ul style="list-style-type: none">Ahorro de tiempo al realizar tareas sencillas sin necesidad de descargar ni instalar ningún programa. (Díaz, 2017, https://www.fuegoyamana.com/blog/aplicacion-web-o-de-escritorio-para-tu-negocio/).	<ul style="list-style-type: none">Perdida de flexibilidad por lo que los usuarios dependen de servidores de terceros. (Maldonado, 2016, p. 13)
<ul style="list-style-type: none">Acceso inmediato es decir que no necesitan ser descargadas, instaladas y configuradas para poder manipularlas. (Díaz, 2017, https://www.fuegoyamana.com/blog/aplicacion-web-o-de-escritorio-para-tu-negocio/).	<ul style="list-style-type: none">Necesariamente se debe mantener conexión a la red. (Aplicaciones Web, 2016, https://lasaplicacioneswebblog.wordpress.com/2016/10/02/ventajas-y-desventajas-de-las-aplicaciones-web/)
<ul style="list-style-type: none">El sistema web puede ser manipulada por múltiples usuarios en el mismo tiempo. (Maldonado, 2016, p. 13)	
<ul style="list-style-type: none">Los sistemas web son multiplataformas basta con tener un navegador y conexión a red para poderlo manipular. (Neosoft, 2018, https://www.neosoft.es/blog/que-es-una-aplicacion-web/)	
<ul style="list-style-type: none">Los sistemas web tienden a no colgarse y así facilitan la centralización de datos. (Maldonado, 2016, p. 13)	

Realizado por: Adriana Quille, 2019.

En este trabajo de titulación se ha decidido implementar un sistema web, a medida que la tecnología va avanzando continuamente y actualmente el internet es uno de los medios tecnológicos más utilizados por los clientes, emprendedores y trabajadores de la fundación; un sistema web hace uso del internet permitiendo el acceso a través de un navegador web debido a que se encuentra alojado en un servidor, ayudando así ahorrar espacio en las computadoras a diferencia de las aplicaciones de escritorio que deben estar previamente instaladas, la característica principal que desarrolla un sistema web es que se puede trabajar desde cualquier lugar es decir sin la necesidad de algún componente externo e independientemente del sistema operativo que se encuentre instalado en la computadora de la cual va acceder.

1.2. Sitio web orientado a los Emprendimientos

Un emprendimiento es un conjunto de competencias que parten desde la actitud de la persona para captar oportunidades del entorno lo cual requiere de pensamientos, ideas innovadoras basándose en la creatividad y así generar proyectos en diferentes áreas para beneficio propio y el medio que lo rodea (Quizhpe et al., 2017, p.3).

Las fundaciones son organizaciones no gubernamentales, sin fines de lucro que se impulsan a diferentes entidades a exponer sus propios productos, generación de empleos entre otras actividades a la que se dediquen, todas estas fundaciones tienen su propia historia de cómo han ido creciendo y ayudando a incrementar mejor los recursos económicos a cada una de las personas que forman parte de ella, todas estas organizaciones se basan principalmente en ideas, investigaciones, experimentos propios que cada uno de sus miembros los plantea siendo estos aprobados por un grupo ya calificado para dar paso a la construcción y posteriormente ofrecer a la comunidad.

Actualmente existe una variedad de Fundaciones entre ellas tenemos a la Fundación Impulsar, Heifer Ecuador, Fundación Emprende Ecuador, CreEcuador, también el Ministerio de Industrias y Productividad, Ministerio de Inclusión Económica y Social, Fundación Siembra Futuro , siendo estas las más reconocidas, quienes ayudan a transformar ideas propias, generación de empleos a través de capacitaciones especializadas en cada una de las ramas, apoyo financiero y mentoría ya sea a jóvenes o personas adultas que tengan deseos de llegar alcanzar sus objetivos aplicando conocimientos propios y quieran iniciar su propio proyecto para un mejor sustento y sin acceso al crédito formal ni capacitación para el establecimiento.

1.3. Arquitecturas de software

Se define como el arte de proyectar, construir y diseñar aplicaciones informáticas, también define la forma de trabajar en un sistema como es la construcción de nuevos módulos pero también implica distinguir una solución estructurada que satisfaga las necesidades todos los requisitos técnicos y operacionales (Panadero, 2017, <https://devexperto.com/arquitectura-del-software/>).

1.3.1. Características

Según (Cervantes, 2018, <https://sg.com.mx/revista/27/arquitectura-software>), presenta las siguientes características que debe poseer cada arquitectura de acuerdo a la que se vaya a implementar dependiendo del caso de estudio que se vaya a realizar.

- ✓ Parte del diseño de software
- ✓ Representa alto nivel de la estructura del sistema
- ✓ Pueden incluir patrones que supervisen la composición de sus componentes.
- ✓ La planificación de proyectos, análisis de riesgos, organización y procesos de desarrollo son los que garantizan la calidad del sistema a desarrollar.
- ✓ Los atributos de calidad son parte de los requerimientos no funcionales.

- ✓ Una arquitectura errónea puede llevar a problema incontables.

Entre las diversas arquitecturas utilizadas con más frecuencia como lo menciona (Ingeniería et al. 2018, pp. 12-16), tenemos las arquitecturas monolíticas, cliente-servidor y 3 capas.

Arquitectura Cliente-Servidor

Es una arquitectura de procesamientos cooperativos, donde uno de los componentes pide servicios a otro, su terminología describe la manipulación entre dos o más programas: una aplicación y un servicio soportante, ofreciendo así al usuario final un acceso transparente a las aplicaciones, datos, servicios de cómputos o cualquier otro recurso del grupo de trabajo a través de múltiples plataformas (Gonzalez, 2018, pp. 3-4).

Arquitectura N capas

Es una arquitectura utilizada por sistemas principalmente que hacen uso de un modelo de negocio, sin embargo, no es recomendable usarla en sistemas de tiempos real; esta arquitectura presenta un diseño de capa intermedia en el proceso, cada capa es un proceso separado y definido corriendo en plataformas separadas, es decir divide la capa de presentación, capa de negocio y la capa de datos (Gómez, 2015, <https://instintobinario.com/arquitectura-en-tres-capas/>).

1.3.2. Comparativa de las Arquitecturas de software

A continuación, se realiza una tabla comparativa entre las diversas arquitecturas utilizadas con más frecuencia como lo menciona (Ingeniería et al. 2018, pp. 12-16), como lo puede observar en la **Tabla 2-1**.

Tabla 2-1: Comparativa de arquitecturas

	Cliente-Servidor	N capas
Protecciones y privilegios	✓	✓
Rapidez de ejecución	✓	✓
Flexibilidad	✓	✓
Escalable	✓	✓
Tecnologías maduras y robustas	✓	✓
Mantenibilidad	✓	✓
Reutilización de código	✓	✓
Manejo de errores	-	✓

Realizado por: Adriana Quille, 2019.

Una vez realizado el análisis entre las diversas arquitecturas de software se concluye que la arquitectura N capas es la más eficiente para ser aplicada en cualquier sistema web a desarrollar, sin embargo, por requerimientos de la fundación se ha tomado la arquitectura cliente/servidor para el desarrollo de este trabajo de titulación.

1.4. Arquitectura cliente/servidor

1.4.1. Definición

La arquitectura cliente – servidor consiste en que un cliente realiza peticiones a otro programa (servidor) que le da respuesta (Valle y Gutierrez, 2016, p.1). La arquitectura cliente – servidor **Figura 2-1**, está diseñada para la tecnología Web que ayuda a los usuarios a programar en línea. Esta arquitectura consta de 2 elementos fundamentales que son Cliente y Servidor (Picucci et al., 2015, p.3).

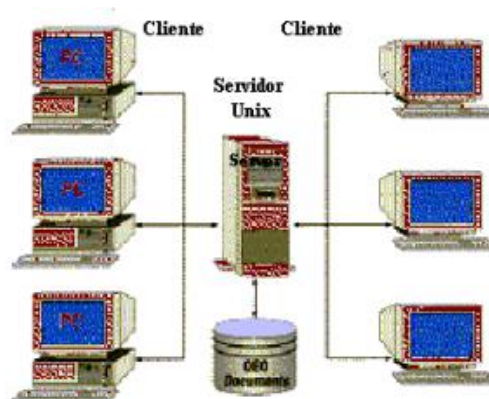


Figura 2-1: Arquitectura cliente – servidor
Fuente: (Valle y Gutierrez, 2016, p.4)

Cliente

Aquel que participa activamente en las conexiones a través de redes LAN o WAN, es decir que envía peticiones al servidor y espera una respuesta, el tiempo de vida es finito cada vez que concluye con las peticiones el trabajo de esta termina (EcuRed, 2016, https://www.ecured.cu/Arquitectura_Cliente_Servidor).

Servidor

Equipo que ofrece varios servicios al cliente por medio de las redes LANs o WANs como por ejemplo la manipulación de archivos, impresiones, páginas web, actualizaciones de BD, controles de acceso, entre otros (Cubillos, 2015, p.2). A continuación, se observa de manera gráfica la arquitectura cliente – servidor.

1.4.2. Características

A continuación, presenta las siguientes características que debe ofrece la arquitectura cliente-servidor:

- La arquitectura cliente – servidor actúa en un solo ente y también entre varios entes realizando todas las peticiones que el cliente lo ha realizado (Bysslender, 2016, <https://programacionviiulatbysslender.wordpress.com/2016/05/24/caracteristicas-de-la-arquitectura-cliente-servidor/>).
- La escalabilidad que presenta la arquitectura cliente – servidor permite que en cada una de las plataformas puedan realizar cambios ya sea por el lado del cliente o del servidor, es decir que las actualizaciones que se realicen serán actualizadas de forma transparente hacia el Cliente que lo esté manipulando (M.I.D Salazar, 2016, http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro21/42_arquitectura_clienteservidor.html).
- La arquitectura cliente – servidor permite interactuar con el cliente de forma inmediata a través de mensajes, donde el mensaje es él envió y la entrega de solicitud del servicio (Saavedra, 2016, <https://programacion7ulatsaavas.wordpress.com/2016/05/24/caracteristicas-arquitectura-cliente-servidor>).
- Las funciones de la arquitectura cliente – servidor pueden ser dinámicas, así como convertirse en cliente mientras realiza las peticiones de las solicitudes enviadas en otras plataformas, pero dentro de una misma red (Microsoft, 2016, p.5).

1.4.3. Ventajas y desventajas

La arquitectura Cliente – Servidor presenta las siguientes ventajas y desventajas que se muestran a continuación, en la **Tabla 3-1**:

Tabla 3-1: Ventajas y desventajas de la arquitectura cliente/servidor.

Ventajas	Desventajas
<ul style="list-style-type: none">• Ahorro de dinero cuando se realiza las migraciones de mainframe a cliente/servidor los mainframe son costosos, su tiempo de desarrollo es menor y son reutilizables tanto en back – end como en front – end reduciendo costos de mantenibilidad y desarrollo; pero aumentando productividad (Flores, 2018, p.8).	<ul style="list-style-type: none">• Número excesivo de clientes ocasionan problemas al servidor provocando congestión en el tráfico de interacción (Lizama, Kindley y Jeria, 2016, p4).

<ul style="list-style-type: none"> Control centralizado existía una descentralización de información ocasionada por la seguridad ocasionando dificultad en la manipulación de la información por lo que ahora la arquitectura se encarga de la implementación de 2 tipos de seguridad el cliente y el servidor. (Flores, 2018, p.11) 	<ul style="list-style-type: none"> En caso de que el servidor o la internet presente fallos, todos los clientes presenciaran de estas molestias (Bysslender, 2016, https://programacionviiulatbysslender.wordpress.com/2016/05/24/caracteristicas-de-la-arquitectura-cliente-servidor/).
<ul style="list-style-type: none"> Flexible y Escalable debido al exceso de las solicitudes es necesario realizar las migraciones o a su vez agregar nuevas funcionalidades, pero sin afectar las operaciones de las instituciones (Flores 2018, p.9). 	

Realizado por: Adriana Quille, 2019.

Después del análisis realizado con las arquitecturas de software, se determina que la arquitectura cliente – servidor es la que se implementará para el desarrollo del presente sistema web debido a que los clientes tienen la libertad de obtener la información que requieran en un momento dado para ser procesada como según le convenga.

1.5. Patrón de Diseño MVC (Modelo Vista Controlador)

1.5.1. Definición

Es un patrón de diseño de software encargado de separar de una aplicación la lógica de negocio de la interfaz de usuario, este patrón de diseño es el más utilizado para el desarrollo de sistemas web también presenta características de alto nivel facilitando mantenimiento, funcionamiento y la respectiva escalabilidad del sistema (Bahit, 2016, p.36).

MVC es aquel que nos facilita la reutilización de código, facilitando el desarrollo de las aplicaciones y para su respectivo mantenimiento (Arcos-Medina, Menéndez y Vallejo, 2018, p.244).

1.5.2. Elementos MVC

El patrón de diseño MVC cuenta con tres módulos diferentes cada uno de estas con funcionalidades correspondientes y bien definidas, siendo estas:

- **Modelo:** es el módulo que contiene la lógica de negocios, es decir es el medio encargado de acceder a los datos de manera directa como intermediario con la base de datos (Bahit, 2016, p.36).

- **Vista:** módulo encargado de mostrar al usuario la forma gráfica del sistema (Bahit, 2016, p.36).
- **Controlador:** módulo encargado del control de peticiones que se lo realiza mediante el modelo, entregando la respuesta a la vista para que sea visualizado por el usuario (Bahit, 2016, p.36).

1.5.3. Labor del patrón MVC.

El funcionamiento de este patrón según (Bahit, 2016, p.37) se basa en:

1. El usuario realiza la petición.
2. El controlador recibe la petición a través de los eventos.
3. Hace el respectivo llamado al modelo.
4. El modelo empieza a interactuar con la base de datos para posteriormente proporcionar información al controlador.
5. El controlador recibe la información y la pasa a la vista.
6. Por último, la vista es la encargada de procesar la información que ha recibido y presentar al usuario.

A continuación, se presenta la **Figura 3-1**, que describe el proceso indicado.

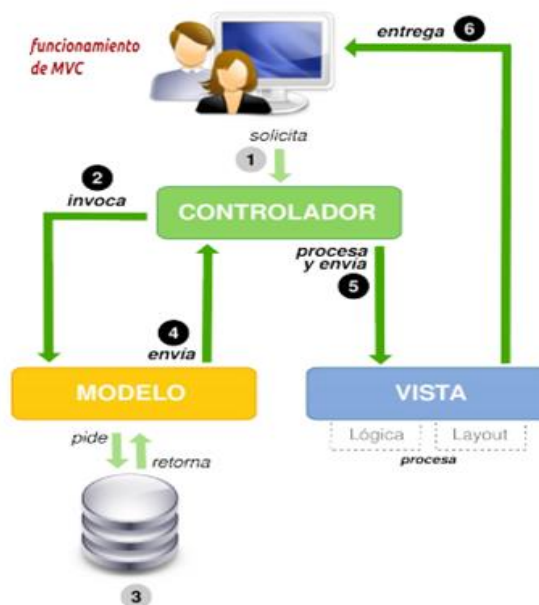


Figura 3-1: Arquitectura MVC
Fuente: (Bahit, 2016, p.37).

1.5.4. Ventajas y desventajas del patrón MVC

Entre las ventajas y desventajas del patrón MVC encontradas, presentamos en la **Tabla 4-1**, tenemos las siguientes:

Tabla 4-1: Ventajas y desventajas de MVC.

Ventajas	Desventajas
<ul style="list-style-type: none">Fácil adaptabilidad a nuevas interfaces de usuario (León et al., 2016, p.565).	<ul style="list-style-type: none">MVC requiere una implementación de gran cantidad de clases entorno al desarrollo, para la etapa de mantenimiento es mucho más mantenible, extensible y modificable a implementarlo (Bascón, 2004, pp.5-6).
<ul style="list-style-type: none">Permite generar o sustituir componentes de las Interfaces permitiendo presentar información actualizada (Bahit, 2016, p.37).	<ul style="list-style-type: none">MVC es un patrón diseño orientado a objetos lo que ocasiona grandes costos y por parte del programador se le hace difícil cuando este no cuenta con el paradigma que él lo realiza (Bascón, 2004, pp.5-6).
<ul style="list-style-type: none">MVC permite realizar modificaciones como aumentar métodos o contenidos los mismos que pueden realizar en el módulo respectivamente es decir sin ocasionar problemas a los demás módulos (Bascón, 2004, p.5).	

Realizado por: Adriana Quiille, 2019

El patrón MVC fue el seleccionado para el desarrollo del sistema web ofreciendo una mejor estructura separando la interfaz del controlador y el modelo. Además, este patrón facilitará la mantenibilidad del sistema cuando se presente modificaciones a las interfaces ya existentes.

1.6. Lenguajes de Programación orientados a la web

Programación es un arte que consiste en ingresar a la computadora una secuencia de pasos para lograr con el objetivo planteado, es decir la computadora cumple ordenes que el ser humano lo requiera (Sánchez y Cano, 2014, p. 2).

Actualmente existe una diversidad de lenguajes de programación modernos, los mismos que pueden ser clasificados en dos grupos, siendo estos de bajo nivel definiendo que es el lenguaje que la máquina entiende y el de alto nivel que se encuentra desarrollado a base de órdenes, instrucciones, sintáxis; que hacen que sea entendido por el ser humano (Trujillo, 2017, <http://www.mods.la/blog/lenguajes-de-programacion-que-son-y-para-que-sirven/>). Entre los lenguajes de alto nivel se puede mencionar PHP, Python, Ruby, ASP (Sánchez y Cano, 2014, p. 3).

PHP

Lenguaje de programación con alto nivel de código abierto, desarrollado para la creación de sitios web, ejecutado por lado del servidor su sintáxis está basada en lenguaje C; también puede ser incrustado en HTML, es decir que en un mismo archivo se podrá combinar código PHP y HTML siendo las respectivas reglas de codificación (Pérez, 2007, <http://www.maestrosdelweb.com/los-diferentes-lenguajes-de-programacion-para-la-web/>).

PYTHON

Lenguaje programación ágil, sencillo y muy corto para aprender, basada principalmente en profesionales estadísticos para generar gráficos con grandes datos, en donde las empresas incorporan back – end y el front – end (BBVAOPEN4U, 2016, <https://bbvaopen4u.com/es/actualidad/ventajas-e-inconvenientes-de-python-y-r-para-la-ciencia-de-datos>).

RUBY

Lenguaje de programación interpretado de alto nivel, requiere de un framework para su funcionamiento, diseñado principalmente para crear todo tipo de aplicaciones, independiente de la plataforma, es decir que no necesariamente debe ser compilado el código fuente para que este sea ejecutado (García, 2017, <https://openwebinars.net/blog/que-es-ruby/>).

ASP

Lenguaje de programación orientado a desarrollar sitios web dinámicos, el mismo que para ser ejecutado se deberá tener instalado Internet Information SERVER (IIS), además se puede combinar código HTML, scripts y componentes ActiveX del servidor (Pérez, 2007, <http://www.maestrosdelweb.com/los-diferentes-lenguajes-de-programacion-para-la-web/>).

1.6.1. Comparativa entre diferentes lenguajes de programación orientados a la web

A continuación, se realiza una tabla comparativa entre los lenguajes de programación de alto nivel para el desarrollo del sistema web, **Tabla 5-1**.

Tabla 5-1: Comparativa de lenguajes de programación.

Lenguaje	PHP	PYTHON	RUBY	ASP
Lenguaje rápido y fácil de aprender	Alta	Escasa	Alta	Alta
Velocidad	Alta	Baja	Media	Media
Multiplataforma	Alta	Alta	Alta	Alta

Tipo de variables	Alta	Sin definir	Baja	Sin definir
Orientado a Objetos	Alta	Alta	Alta	Baja
Utiliza librerías	Alta	Alta	Baja	Alta
Seguridad	Alta	Lenta y escasa	Escasa	Escasa

Realizado por: Adriana Quille, 2019.

Se decidió utilizar PHP para el desarrollo del sistema web debido a que es un lenguaje fácil de aprender y existe gran variedad de documentación ante cualquier duda o problema que puede presentarse durante el proceso de desarrollo. Además, PHP es multiplataforma y compatible ante cualquier gestor de base de datos.

1.7.PHP

1.7.1. Definición

PHP (Hypertext Pre-processor) lenguaje orientado a los Script del lado del Servidor. Estos Scripts se comunican con los documentos HTML y el servidor es el encargado de interpretar y ejecutar antes de enviar las paginas al cliente, es decir el proceso se lo realiza de manera transparente y el cliente observara los resultados finales (González, 2015, p.3), como indica en la **Figura 4-1**.



Figura 4-1: Arquitectura PHP.

Fuente: (Maraboli, 2003, p.2).

1.7.2. Características

- **Variedad de herramientas:** existe una variedad de herramientas que PHP ofrece a los desarrolladores se puede contratar un servidor para cargar los archivos de la aplicación como también se podrá instalar una máquina virtual o a su vez descargar herramientas libres como WAMP, XAMPP siendo herramientas que ayudan a simular un servidor web para ejecutar las operaciones que se realicen (Miró, 2016, p.1).

- **Velocidad:** PHP ejecuta de manera rápida y crea retrasos en la máquina permitiendo no generar grandes recursos para la aplicación, lo que le hace diferentes es que PHP se integra con ambientes Unix que hacen que la aplicación sea más exitosa (Gráfica, 2016, <http://redgrafica.com/El-lenguaje-de-programacion-PHP>).
- **Estabilidad:** PHP ofrece una administración de recursos sofisticado en el manejo de variables lo que ayuda al sistema a ser más estable y robusto (Gráfica, 2016, <http://redgrafica.com/El-lenguaje-de-programacion-PHP>).
- **Seguridad:** posee una diversidad de niveles de seguridad las mismas que serán implementadas de acuerdo a las necesidades de la aplicación en desarrollo, todo este proceso se deberá realizar en el archivo **.ini** (Gráfica, 2016, <http://redgrafica.com/El-lenguaje-de-programacion-PHP>).
- **Programación orientada a objetos:** divide los scripts en métodos, clases, y entre otros para que la aplicación sea más rápida al emitir resultados, también divide la estructura en Modelo Vista Controlador (Miró, 2016, p.2).
- **Módulos externos para mejorar las aplicaciones webs:** permite incorporar funcionalidades ya existentes de acuerdo a las necesidades del programador, todo esto es posible realizar con PHP porque cuenta con una estructura MVC (Miró, 2016, p.2).

1.7.3. Ventajas

Entre las ventajas que PHP brinda, se puede mencionar las siguientes en la **Tabla 6-1:**

Tabla 6-1: Ventajas de PHP.

Ventajas
<ul style="list-style-type: none"> • PHP es un lenguaje multiplataforma es decir libre (Miró, 2016, p.2).
<ul style="list-style-type: none"> • PHP posee una sintáxis parecida a C, permitiendo que cualquiera lo pueda manipular y entenderla rápidamente (Gráfica, 2016, http://redgrafica.com/El-lenguaje-de-programacion-PHP).
<ul style="list-style-type: none"> • PHP se encuentra orientado al desarrollo de aplicaciones web dinámicas con un almacenamiento de información en una base de datos (Achour et al., 2003, pp.7-8)
<ul style="list-style-type: none"> • PHP soporta una gran cantidad de base de datos también cuenta con extensiones que le permiten usar de manera transparente cualquier BD soportada con la extensión en uso (Achour et al., 2003, p.9).
<ul style="list-style-type: none"> • En la actualidad la mayoría de las Webs que son complejas trabajan con PHP por lo que las empresas requieren de desarrolladores especializados en PHP (Miró, 2016, p.2).

Realizado por: Adriana Quille, 2019.

1.8. Framework de Desarrollo

Un framework es una herramienta la cual permite la reutilización de código orientado a objetos, modela las relaciones generales de diversas entidades de un determinado dominio; encargado también de describir objetos que componen el sistema y su manera de interactuar ante sus interfaces, flujos de control y las responsabilidades que poseen cada uno de los objetos (Navarrete y Navarro, 2017, p.2).

1.8.1. Definición

Un framework de desarrollo es un conjunto de librerías, herramientas y normas a seguir que ayudaran al desarrollo de las aplicaciones, estos framework son desarrollados por programadores de sistemas, quienes permiten una reutilización de código, la estandarización del desarrollo y la utilización del ciclo de desarrollo de tipo interactivo para mejorar la codificación y posteriormente una mantenibilidad a su evolución (Lafosse, 2010, pp.12-13).

Un framework también permite agregar funcionalidades extendidas a un lenguaje de programación, trabaja con la mayoría de los patrones de programación para dar solución a lo propuesto, mejorándole y haciendo que sea lo más entendible posible y sostenible permitiendo separar la aplicación en capas (Camacho, Biancha y Daniel, 1995,p.178).

1.8.2. Ventajas y Desventajas

Para utilizar un framework basado en PHP para el desarrollo del software según (Suarez, 2012, <https://www.kabytes.com/programacion/usar-o-no-un-framework/>) presentan las siguientes ventajas y desventajas en la **Tabla 7-1.**:

Tabla 7-1: Ventajas y desventajas de los framework

Ventajas	Desventajas
<ul style="list-style-type: none">• Un framework orientado a PHP se encuentra basada principalmente en los estándares que permiten la compatibilidad con la mayoría de los navegadores.	<ul style="list-style-type: none">• Todo los framework requieren de un tiempo de aprendizaje, programar al principio suele ser tedioso y cuesta aprenderlo, pero existen manuales o cursos de ayuda para lograr este objetivo.
<ul style="list-style-type: none">• Permite crear flujos estandarizados de trabajo es decir que el equipo de desarrollo pueda entender el código sin ningún problema el mismo que se encuentra codificado de manera fácil y sencillo de entenderlo.	<ul style="list-style-type: none">• Desconocimiento total de cómo funciona internamente un framework, pero esta desventaja es rara vez que se debe realizar alguna modificación.

<ul style="list-style-type: none"> • Permiten la incorporación de diversas librerías plugin y ayudas para la comunidad de desarrollo la misma que permite la reutilización de código en caso de ser necesaria. 	
<ul style="list-style-type: none"> • Seguridad es uno de los temas principales que cuenta cada uno de los framework protegiendo lo que es sesiones, url, formularios y las respectivas consultas de la base de datos. 	

Realizado por: Adriana Quille, 2019.

1.8.3. Tipos de Framework para PHP

Existe una variedad de framework basados en PHP, pero entre los principales competidores tenemos a Symfony, Laravel, Zend Framework y CakePHP como lo menciona (Camacho, Biancha y Daniel, 1995,p.178).

Symfony

Es un framework para PHP que permite construir diversos componentes de manera independiente, pero todos estos creados por el proyecto Symfony, además posee un conjunto de librerías que se utilizan para la creación de aplicaciones PHP (Symfony, 2019, <https://symfony.es/pagina/que-es-symfony/>).

Laravel

Laravel es un framework para entorno web de código abierto que manipula la orientación a objetos para el desarrollo de aplicaciones web, con el fin de emplear una experiencia agradable y satisfactorio (Carvajal, 2016, p.3).

Zend Framework

Es un framework de código abierto, orientado a objetos y desarrollado para aplicaciones y servicios web que permite la adaptación a cualquier proyecto, pero de difícil comprensión y aprendizaje (Technologies, 2017, p.7).

CakePHP

Es un framework de gestión de sesiones con librerías independientes para PHP, de código abierto, ofrece una estructuración y rapidez sin perder la flexibilidad, proporciona una caja de herramientas es decir todo en uno (Cake Software Foundation, 2019, p.7).

1.8.4. Comparativa de Framework para PHP

A continuación, en la **Tabla 8-1**, se visualiza varias características de los principales frameworks para PHP siendo estos Symfony, Laravel, Zend Fw, CakePHP (Andino, 2016, pp.14-15).

Tabla 8-1: Comparativa de Framework

CARACTERÍSTICAS	Symfony	Laravel	Zend Fw	CakePhp
<i>Arquitectura de Aplicaciones</i>				
Incorporan el patrón MVC orientado a objetos.	✓	✓	✓	✓
Operaciones CRUD asociados al Active Record.	✓	✓	-	✓
Mapeados de objetos a las BD relacionales, conocidas como ORM.	✓	✓	-	✓
<i>Acceso Web</i>				
Resuelve peticiones HTTP.	✓	✓	✓	✓
Capaz de generar URLs amigables.	✓	✓	✓	✓
<i>Implementación De Código HTML</i>				
Permite hacer uso de las plantillas PHP.	✓	✓	✓	✓
También se puede hacer uso de la plantilla Twig.	✓	-	-	-
Facilita incorporar ayudantes de plantillas.	✓	-	-	✓
<i>Seguridad</i>				
Manejo personalizado de Sesiones para los usuarios.	✓	✓	-	✓
<i>Usabilidad Y Acceso Rápido</i>				
El almacenamiento lo realiza en la caché de las vistas	✓	✓	-	✓
El almacenamiento lo realiza en la cache de configuración de las aplicaciones.	✓	✓	-	-
<i>Documentación Para Su Uso</i>				
Manuales de referencia actualizadas en la web.	✓	✓	✓	✓
Documentación de la Interfaz de programación de aplicaciones conocidas como API.	✓	✓	✓	✓
<i>Herramientas de Programación</i>				
Genera código PHP.	✓	✓	-	✓
Proporciona herramientas para depuraciones y pruebas.	✓	-	✓	-
Ofrece una línea de comandos para la creación y mantenimiento de las aplicaciones.	✓	✓	-	✓
<i>Extensibilidad Y Opciones Adicionales</i>				
Permite incorporar otras herramientas a través de los plugins.	✓	-	-	✓

Realiza la implementación a través de Asynchronous JavaScript y XML (Ajax).	✓	✓	-	✓
Permite enviar correos electrónicos.	✓	✓	✓	
Permite generar archivos PDF	✓	✓	✓	✓
Características Adicionales				
Cuenta con licencias libres	✓	✓	✓	✓
Mantiene una conectividad activa de usuarios.	✓	✓	-	✓

Fuente: (Tupe y Cisneros, 2008, pp.3-8).

Realizado por: Adriana Quille, 2019.

Del análisis comparativo entre los frameworks para PHP se determina que Symfony brinda mayores características que los otros, siendo el óptimo para el desarrollo del sistema web.

1.9.Symfony

Symfony surge en el 2003 cuando Fabien realiza una investigación con las herramientas de software libre disponible para el desarrollo de aplicaciones web con PHP.

Symfony es el framework más completo que existe en la actualidad, está diseñado para la optimización, separando la lógica de negocio, la lógica del servidor y la presentación del sistema web, este framework ofrece una variedad de herramientas enfocadas para la reducción de tiempos en el desarrollo del sistema ya sea lo más complejo posible (Potencier y Zaninotto, 2008, P.13).

1.9.1. Características

Symfony fue diseñado para que se ajustara a los requisitos que se presenta a continuación:

- Fácil de instalar y configurar en la mayoría de plataformas independientemente del sistema gestor de bases de datos (Potencier y Zaninotto, 2008, P.13).
- Se puede implementar en cualquier caso por más complejo que sea gracias a la flexibilidad de adaptarse a cualquier entorno (Potencier y Zaninotto, 2008, P.13).
- Symfony está preparado para sistemas empresariales y adaptable a las políticas y arquitecturas de cada uno de estas permitiendo una estabilidad a largo plazo (Potencier y Zaninotto, 2008, P.13).

- Symfony permite realizar modificaciones o implementar librerías que han sido desarrollados por terceros sin ocasionar ningún problema (Potencier y Zaninotto, 2008, P.13).
- Symfony se encuentra desarrollado bajo la arquitectura MVC donde todas las clases y archivos lo relaciona con el modelo, por otra parte, las plantillas que se almacenan se guardan en diversos directorios repartidos por el proyecto (Andino, 2016, pp.14-15).

1.9.2. Estructura de directorios en Symfony

Symfony posee una estructura de directorios puede ser modificada de acuerdo con la programación que se esté desarrollando a continuación, en la **Figura 5-1**, se observa la estructura de directorios que Symfony ofrece.

Estructura de directorios de Symfony 4

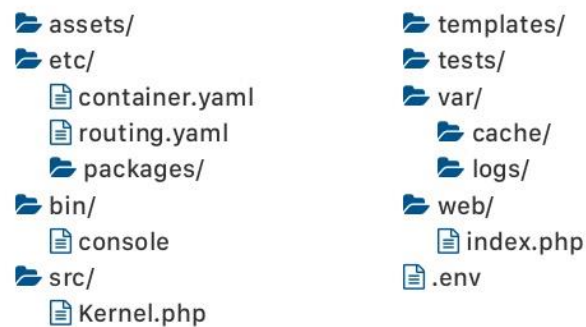


Figura 5-1: Estructura de directorios de Symfony 4
Fuente: (Eguiluz, 2017, p.71).

- **Etc/:** contiene la configuración de la aplicación.
- **Src/:** contiene todo el código PHP de la aplicación.
- **Tests/:** contiene los directorios de prueba.
- **Var/:** contiene los directorios de los logs.
- **Web/:** directorio raíz que contiene todos los archivos a los que se acceder.

Symfony tiene como particularidad automatizar las tareas comunes para la gestión de datos (CRUD) permitiendo al desarrollador dedicarse por completo a los aspectos específicos que el sistema web requiere obteniendo funcionalidades exitosas y ahorro de tiempo.

1.9.3. Implementación de MVC que realiza Symfony

En la **Figura 6-1**, puede identificar como Symfony hace uso de la arquitectura MVC que trae incorporada por defecto este framework.

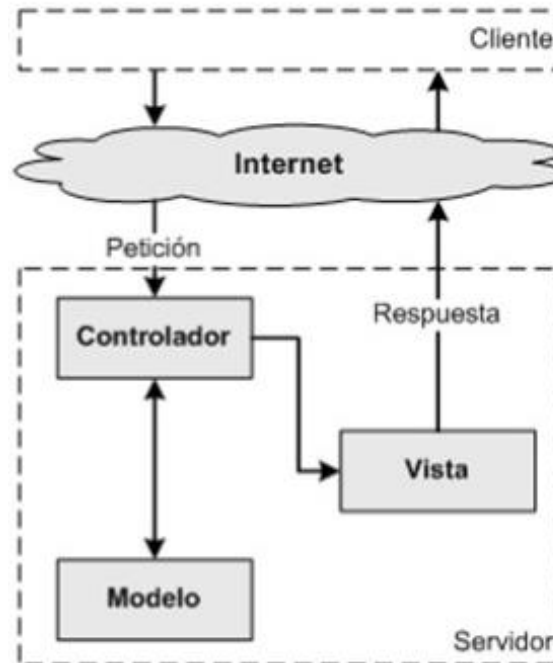


Figura 6-1: Arquitectura MVC en Symfony

Fuente: (Acosta, Ceballos y Rodallega, 2014, p.12)

La implementación de Symfony se acoge con el patrón de diseño MVC ocasionando que la aplicación sea más rápida y sencilla. Deberá constar de un controlador frontal que es aquel que mantiene código relativo a MVC el mismo que es generada automáticamente por Symfony. También se hace uso de la librería Propel que está encargada de generarse automáticamente permitiendo así crear el esqueleto y estructura básica de las clases y el código necesario; por otra parte la lógica de vista se convierte en un archivo de configuración sin tener la necesidad de programarla (Potencier y Zaninotto, 2008,pp.32-33).

1.10. Bootstrap

1.10.1. Definición

Bootstrap es un framework orientado al usuario para la creación de páginas web dinámicas y aplicaciones web, facilitando el procesamiento rápido y sencillo, es compatible con todos los navegadores principales como Internet Explorer, Google Chrome, Opera Firefox y Safari; carga de manera inmediata las páginas web. Bootstrap cuenta con plantillas diseñadas en HTML y CSS

para varios componentes de la interfaz, permitiendo el acceso a temas, menús, diseños, botones, entre otros, (Bharat, 2018, <https://vmokshagroup.com/blog/bootstrap-advantages/>).

1.10.2. Ventajas y desventajas de Bootstrap

Según (Bharat, 2018, <https://vmokshagroup.com/blog/bootstrap-advantages/>), Bootstrap nos presenta las siguientes ventajas y desventajas expuestas en la **Tabla 9-1**.

Tabla 9-1: Ventajas y desventajas de Bootstrap.

Ventajas	Desventajas
<ul style="list-style-type: none"> • Ofrece una velocidad de desarrollo, con la codificación ofrece menos funcionalidades CSS y bloqueos de código preconstruídos en lugar de estructurar el código desde cero. 	<ul style="list-style-type: none"> • Para hacer uso de funcionalidades extras se debe utilizar jQuery y JavaScript.
<ul style="list-style-type: none"> • Permite una sensibilidad de equipo con un diseño receptivo y un sistema de cuadrícula de 12 columnas es decir facilita ajustar dinámicamente el sitio web a una resolución de pantalla en el que se encuentre. 	<ul style="list-style-type: none"> • Si desea agregar nuevos componentes o atributos a Bootstrap se lo puede realizar de manera manual, pero manteniendo la relación con el diseño.
<ul style="list-style-type: none"> • Mantiene una consistencia independientemente del diseñador/desarrollador, además los resultados pueden visualizarse en varios navegadores y la salida sigue siendo la misma. 	
<ul style="list-style-type: none"> • Bootstrap permite personalizar los diseños, seleccionar cualquier función de la web según las necesidades del desarrollador. 	
<ul style="list-style-type: none"> • Ayuda a dar solución y soporte a los problemas que se presenta. 	

Realizado por: Adriana Quille, 2019.

Con el objetivo de facilitar el desarrollo de la aplicación se decidió utilizar Bootstrap el mismo que presenta una gran velocidad en el desarrollo presentando códigos prediseñados que ayudan a mejorar las interfaces de usuario, dado un diseño representativo ante cualquier tipo de resolución de pantalla y poderla visualizar en cualquier navegador. Además, en caso de presentar problemas posee de una gran variedad de documentación que ayudara con el soporte necesario para la resolución.

1.11. Tecnología para el Desarrollo de Mensajería

La mejor forma de mantener a los clientes es mediante una respuesta inmediata y para aquello que mejor agregar un chat online al sitio web, esta herramienta servirá para responder de forma directa las consultas online y en el momento exacto como el cliente lo desee (Grupo, 2019, <https://www.grupoeclipse.net/chat-online/index.html>).

Las tecnologías para el desarrollo de mensajería son un punto intermedio entre los sistemas de chat y los correos electrónicos, esta herramienta es la más utilizada en la actualidad por lo que se le considera como el motor que mueve hoy en día a las redes sociales y porque no ahora en el sistema web a desarrollar (Matsushita et al., 2000,p.33).

1.11.1. Definición de mensajería instantánea

Es un tipo de comunicación sincrónica en tiempo real, que funciona principalmente con la conexión a internet y una suscripción a una cuenta de usuario para poseer al servicio de mensajería instantánea (Matsushita et al., 2000,p.35).

1.11.2. Web Socket

WebSocket es un protocolo *ws* que permite crear un canal de comunicación continua full-duplex bidireccional sobre una sola conexión TCP, es decir que permite establecer conexiones con los socket, es decir entre el navegador y el servidor, una vez que ya se estableció la conexión se enviarán directamente los datos a través de un socket en lugar de las respuestas HTTP habituales (Roa, 2016, <https://medium.com/@alejandroroa/websocket-y-php-socket-3d8ac636a108>).

1.11.3. Ventajas y Desventajas de WebSocket

Según (Víctor Díaz Marco, 2015, <https://v0ctor.me/websocket>) presenta al WebSocket con las siguientes ventajas y desventajas:

Ventajas

- Permite establecer conexiones bidireccionales en tiempo real en la web.
- Reduce la saturación de cabeceras.
- Utiliza una solución de integración diferente ante los problemas de compatibilidad.
- Funcionamiento sencillo es decir establece la conexión envía/recibe mensajes y la conexión se cierra.

- Evita problemas de cortafuegos que se encuentra trabajando a través del puerto HTTP.

Desventajas

- El número máximo de conexiones simultaneas que admite el puerto es de 64.000, además cuando las conexiones se encuentren abiertas requiere de más memoria por parte del servidor.

Entre las diversas tecnologías de Web Socket a las más principales que son PHPWebSocket, Learnboos/Socket.io y jWebSocket.

PHPWebSocket

Es una librería que ofrece comunicación bidireccional que utiliza HTTP como transporte y una capa para beneficios de infraestructura como los proxys , filtrado y autenticaciones, esta librería fue implementada para mantener la eficiencia y la fiabilidad en los sistemas (Sadasivan et al., 2011, p.4).

Learnboost /Socket.io

Es una librería open source que permite manejar eventos en tiempo real mediante conexiones TCP y JavaScript, mediante el protocolo *ws*. Esta librería permite mantener una conexión bidireccional y funciona en todas las plataformas, navegadores o dispositivos, su principal característica es que se centra en la confiabilidad y velocidad (Socket, 2019, <https://socket.io/>).

jWebSocket

Es una librería open source para el desarrollo web en tiempo real, es bidireccional lo que significa que empuja del servidor al cliente siendo este compatible para retransmitir mensajes a través de WebSocket, además se encuentra basada en el lenguaje de programación java (Zone, 2011, <https://dzone.com/articles/tomcat-websockets-html5>).

1.11.4. Comparativa de las tecnologías para el chat en línea.

Entre las herramientas más utilizadas en la actualidad para generar un chat en línea como menciona (Hansa, 2010, <https://code.tutsplus.com/es/tutorials/start-using-html5-websockets-today--net-13270>) son PHPWebSocket, Socket.io, jWebSocket; los mismos que se puede visualizar a continuación, en la **Tabla 10-1**.

Tabla 10-1: Tecnologías para chat en línea.

	PHP WebSocket	Learnboost / Socket.IO	jWeb Socket
Tecnología	PHP	Node.js	Java Full-duplex
Arquitectura	Cliente/Servidor	N capas	N capas
Compatibilidad	Responde a través de peticiones HTTP	Responde a través de HTTP los mismos que no ofrecen una solución óptima.	Es extensible y escalable para adaptarse a los requisitos individuales.
Utilidad	Utiliza la seguridad propia de los navegadores	Mejor rendimiento del servidor y mejor experiencia del usuario	Reducción de costos, y alta seguridad en la web
Interoperabilidad	Multiplataforma	Multiplataforma que funciona con versiones tan bajas como es IE 5.5	Multiplataforma y posee una flexibilidad máxima con los navegadores

Realizado por: Adriana Quille, 2019.

Con el análisis realizado entre los diferentes WebSocket que se utilizan para el desarrollo de un chat en línea se decidió utilizar PHPWebSocket, siendo esta tecnología propia del lenguaje de programación PHP; lenguaje con el que se encuentra desarrollado el sistema web, además presenta una curva de aprendizaje baja pero se encuentra familiarizada con el lenguaje antes mencionado, y finalmente porque esta tecnología puede adaptarse a una arquitectura cliente servidor la misma que se emplea para el presente trabajo de titulación.

1.12. Base de Datos

1.12.1. Definición

Base de datos es una herramienta que se utiliza para recopilar y ordenar datos, la misma que puede almacenar todo tipo de información, gran parte de la recolección de datos se lo realiza a través de hojas de cálculo pero a medida que estos datos crecen pierden la productividad y eficiencia por lo que se ven obligados a migrarlos a una base de datos que les ayude a mantener dichos datos de manera ordenada y eficiente, toda la información migrada, permitiéndole manipular la información como se la requiera. Entre los gestores de bases de datos más principales tenemos MySQL, SQL Server, PostgreSQL y Oracle Database (Microsoft, 2019, <https://support.office.com/es-es/article/conceptos-básicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204>).

MySQL

Sistema gestor de base datos basado en software libre conocido por su simplicidad, seguridad y rendimiento, se caracteriza principalmente por la conectividad segura y soporte a las diversas plataformas de almacenamiento, multiusuario y multiplataforma (Pérez et al., 2018,p.235).

SQL Server

Motor de base de dato relacional, permite trabajar de forma nativa ya sea en sistemas Windows o Linux con gran fluidez de información al mismo tiempo, ofreciendo así una escalabilidad, estabilidad y seguridad a sus datos (Microsoft, 2018, <https://www.microsoft.com/es-es/sql-server/sql-server-2017>).

PostgreSQL

Sistema de base de datos basado principalmente en la confiabilidad que presenta al momento de guardar la información, maneja modelo cliente/servidor, también utiliza multiprocesos que ayuda a la escalabilidad del sistema (Todopostgresql, 2018, <https://todopostgresql.com/ventajas-y-desventajas-de-postgresql/>).

Oracle Database

Gestor de base de datos relacional con funciones de lenguaje extensible y utilizado por grandes empresas o multinacionales, este gestor de base de datos proporciona una versión gratuita con la mínima capacidad de almacenamiento orientada a estudiantes, desarrolladores y docentes (Potineni et al., 2018, <https://docs.oracle.com/en/database/oracle/oracle-database/18/admq5/2-day-dba.pdf>).

1.12.2. Comparativa entre bases de datos

A continuación, en la **Tabla 11-1** se presenta una tabla comparativa sobre el estudio de las bases de datos más utilizadas para el desarrollo de sistemas web.

Tabla 11-1: Comparativa de Bases de Datos.

	MySQL	SQL Server	PostgreSQL	Oracle Database
Gratuito y Multiplataforma	Si	Si	Si	Multiplataforma
Optimiza multiprocesamientos confiabilidad y escalabilidad	Si			Si
Soporte Windows, Linux	Si	Windows	Si	Si
Entorno de desarrollo	Java .NET PHP	Java PHP	Java APEX .NET	Java Perl .NET

Transacciones	InnoDB	-	InnoDB	Presenta cuadros regulares y ayuda de operaciones
---------------	--------	---	--------	---

Fuente: (Cachicatari, 2018, <http://www.latindevelopers.com/articulo/diferencias-entre-mysql-y-sql-server/>; Microsoft, 2018, <https://www.microsoft.com/es-es/sql-server/sql-server-2017>; Todopostgresql, 2018, <https://todopostgresql.com/ventajas-y-desventajas-de-postgresql/>; Mejia, 2014, <https://github.com/tatianamejia/Bases-de-Datos/wiki/Bases-de-Datos>).

Realizado por: Adriana Quille, 2019

Después del análisis realizado se utilizó el gestor de base de datos MySQL siendo este un gestor de base de datos de alto nivel y con variedad de información para dar una solución óptima, también es adaptable con el lenguaje de programación de PHP el mismo que se utiliza para el desarrollo del sistema web.

1.12.3. MySQL

MySQL es un gestor de base de datos de código abierto más popular a nivel mundial. Es una de las bases de datos que se ha convertido en la principal opción gracias a su rendimiento probado, a su fiabilidad y confiabilidad de uso para aplicaciones basadas en la web, utilizada por propiedades web de alto nivel como lo es Facebook, Twitter, YouTube (Potineni et al., 2018, <https://docs.oracle.com/en/database/oracle/oracle-database/18/admq/2-day-dba.pdf>).

1.12.4. Características

MySQL es marca comercial de Oracle, tiene doble licencia, los usuarios pueden elegir si utilizar como un producto de código abierto según los términos de licencia GPL o la licencia Comercial estándar (Oracle, 2019, <https://dev.mysql.com/doc/refman/8.0/en/introduction.html>).

Entre las características principales de MySQL tenemos las siguientes (MySQL, 2019, <http://ftp.tsrc.edu.tw/MySQL/doc/refman/5.0/es/features.html>).

- MySQL se encuentra escrito en C y C++, indispensable para trabajar con las APIs disponibles en C, C++, Java, PHP, Ruby.
- Utiliza la arquitectura multihilos mediante los propios hilos del kernel, las mismas que pueden fácilmente ser utilizadas por CPUs disponibles.
- Emplea tablas Hash en memoria, las mismas que son utilizadas como tablas temporales.

- Ofrece una seguridad muy flexible y segura permitiendo la verificación a través de los hosts, también las contraseñas son seguras y cuenta con un sistema de encriptado cuando se conecta con el servidor.
- Permite una conectividad con el servidor MySQL utilizando sockets TCP/IP para cualquier plataforma, y a nivel de cliente hacen uso de los named pipes para su conexión.

1.12.5. Ventajas y desventajas

Según (Enríquez et al., 2017,p.2), MySQL presta las siguientes ventajas y desventajas visualizadas en la **Tabla 12-1**.

Tabla 12-1: Ventajas y desventajas de MySQL.

Ventajas	Desventajas
<ul style="list-style-type: none"> • Proporciona una velocidad al momento de realizar las operaciones, ocasionando mejor rendimiento. 	<ul style="list-style-type: none"> • No es intuitivo como Access.
<ul style="list-style-type: none"> • Posee costos bajos para la construcción de la base de datos debido a su bajo consumo puede ser ejecutado en máquinas con escasos recursos sin ocasionar problemas. 	<ul style="list-style-type: none"> • Algunas herramientas de MySQL no se encuentran documentadas.
<ul style="list-style-type: none"> • Fácil de instalar y configurar, también soporta cualquier sistema operativo. 	
<ul style="list-style-type: none"> • Mantiene una probabilidad baja de corromper datos, y dar solución a los errores ocasionados. 	

Realizado por: Adriana Quille, 2019.

1.13. Servidor Apache

Apache es un software de servidor web gratuito y de código abierto, su nombre original en Apache HTTP Server, permite a los propietarios servir contenidos en la web, es decir cuando necesite visitar algún sitio web deberá ingresar el nombre de dominio en la barra de direcciones del navegador, luego el servidor web es el encargado de enviar los archivos que han sido solicitados actuando como un repartidor virtual (Bustos, 2019, 2019, <https://www.hostinger.es/tutoriales/que-es-apache/>).

Servidor web es un programa diseñado especialmente para transferir datos de hipertexto, como son las páginas web con todos sus elementos haciendo uso del protocolo HTTP, estos servidores deben ser alojados en un ordenador el mismo que debe contar con conexión a internet. El web server actúa de manera inmediata en cuanto recibe alguna petición desde un navegador por

ejemplo, acceder a la página web y responde la petición enviando código HTML mediante una transferencias de datos a través de la red (Bustos, 2019, <https://www.hostinger.es/tutoriales/que-es-apache/>).

1.13.1. Características

Entre las principales características se encuentran las siguientes:

- Realiza autenticación de datos haciendo uso de SGDB.
- Soporta diversos lenguajes como PHP, Python y tcl.
- Posee seguridad SSL y TLS.

1.13.2. Ventajas y desventajas del servidor Apache

Según (Espinoza, 2011, pp.2-3) entre las ventajas y desventajas **Tabla 13-1**, que ofrece el servidor apache tenemos:

Tabla 13-1: Ventajas y desventajas del servidor Apache

Ventajas	Desventajas
<ul style="list-style-type: none"> • Se desarrolla con el protocolo HTTP 	<ul style="list-style-type: none"> • No posee de una buena administración
<ul style="list-style-type: none"> • Es modular 	<ul style="list-style-type: none"> • Tiene formatos de configuración no estándar.
<ul style="list-style-type: none"> • Fácil de conseguir ayuda y soporte 	
<ul style="list-style-type: none"> • Posee código abierto 	
<ul style="list-style-type: none"> • Es multiplataforma 	
<ul style="list-style-type: none"> • Posee licencia freeware y un amplio nivel de capacitación y la compatibilidad con los diferentes sistemas operativos. 	

Realizado por: Adriana Quille, 2019.

Al hacer uso del servidor Apache, se accede al soporte técnico, el mismo que garantizara el correcto funcionamiento del sistema web, evitando ocasionar problemas antes del tiempo establecido para la resolución. Además, Apache es multiplataforma y de código abierto que puede ser instalado sin ningún costo adicional.

1.14. Metodología de desarrollo

Una metodología de desarrollo de software es identificada como la forma de interpretar la realidad para dar solución a un problema, por tanto la metodología de desarrollo consiste en hacer uso de diferentes herramientas, técnicas y modelos para el desarrollo, también es considerada como una forma de planificar y controlar los procedimientos de creación de un problema a resolver (Gomez,

2017, <https://www.megapractical.com/blog-de-arquitectura-soa-y-desarrollo-de-software/metodologias-de-desarrollo-de-software>).

1.14.1. Características

Entre la características que hace mención la metodología de desarrollo de software según (StudentPlace, 2018, <https://studentplace98.blogspot.com/2018/09/metodologia-de-desarrollo-de-software.html>), tenemos las siguientes:

- Define características para llevar a cabo un proyecto
- Relacionar criterios de organización para el desarrollo del proyecto
- Asegurar la calidad del desarrollo del software
- Satisfacer las necesidades del cliente
- Conseguir alto rendimiento y eficiencia del personal asignado al desarrollo
- Ajustar plazos y costos previstos durante el proceso de planificación
- Facilitar mantenibilidad posteriormente

1.14.2. Comparativa de metodologías tradicional vs ágil

A continuación, en la **Tabla 14-1**, se visualiza una comparativa entre las metodologías tradicionales y ágiles, de acuerdo a (Allegue, 2018, <https://lecciondeaprendizaje.blogspot.com/2018/03/comparacion-de-metodologias-agile-scrum.html>), menciona lo siguiente:

Tabla 14-1: Comparativa de metodologías tradicional vs ágil

	Metodología Tradicional	Metodología ágil
Énfasis	Procesos	Persona
Documentación	Se documenta todo	Según lo que se requiera
Estilo de procesos	Lineal	Iterativo
Priorización de los requisitos	Debe ser fijo en el plan de proyectos	Según el valor del negocio la misma que debe ser actualizada
Seguro de calidad	Enfocada en los procesos	Enfocada en el cliente
Cambios	Sistema de gestión de cambio	Las actualizaciones de priorización según el Producto Backlog
Liderazgo	Mando y control	Colaborativo con el líder del servicio
Medición de rendimiento	Plan de conformidad	Valor del negocio
Participación del Cliente	Varía de acuerdo al ciclo de vida del proyecto	Es alta durante todo el proceso de desarrollo del proyecto.

Realizado por: Adriana Quille, 2019.

El presente trabajo de titulación se manipula el sistema web utilizando la metodología ágil, dado que esta metodología presenta priorización de recursos, seguro de calidad, cambios y cuenta con la característica más principal que es la participación con el cliente.

1.15. Metodologías Ágiles

Ágil es un proceso que colabora a los equipos a facilitar respuestas rápidas e imprescindibles ante los diversos cambios que se presenten sobre el proyecto que se encuentre en desarrollo, es decir crea oportunidades para evaluar la dirección de un proyecto durante el proceso de desarrollo. Normalmente los quipos de trabajo evalúan el proyecto mediante reuniones regulares conocidas como iteraciones o sprint (Gonçalves, 2019, <https://luis-goncalves.com/es/que-es-la-metodologia-agil/>).

1.15.1. Características

Según (Martinez, 2018, <https://superrhheroes.sesametime.com/la-metodologia-agil/>) las características principales de utilizar una metodología ágil son:

- Satisfacer al cliente a través de las entregas tempranas y continuas del software funcional, en periodos de un par de semanas hasta un par de meses.
- Los requisitos cambiantes son bienvenidos, aunque retrasen el desarrollo del software, pero permitiendo obtener una ventaja competitiva con el cliente.
- Trabajar en conjunto personal del negocio y desarrolladores para una buena construcción de proyectos.
- La comunicación directa es la forma más eficiente para transmitir ideas de ida y vuelta.

1.15.2. Comparativa de metodologías ágiles

A continuación, se presenta una tabla comparativa **Tabla 15-1**, sobre el estudio de las metodologías ágiles más utilizadas en los últimos años siendo estas: SCRUM, Programación Extrema (XP) y KANBAN según (Hmd Project Managers, 2016, <https://uv-mdap.com/blog/cuales-son-las-reglas-del-kanban-algunos-tips-y-reflexiones/>).

Tabla 15-1: Comparativa de metodologías ágiles.

	SCRUM	XP	KANBAN
Estrategia de desarrollo iterativo e incremental	Si	Si	Incremental
Retroalimentación continua entre el equipo de desarrollo y el cliente	Si	Si	Escasa
Calidad de resultado	Si	Si	Si

Solapamiento de las diversas fases de desarrollo	Si	Si	-
Reuniones de planificación al inicio de cada Sprint (15min)	Si	Si	Se realiza de manera iterativa
Reuniones al finalizar cada sprint para conocer las tareas exitosas y fallidas	Si	Si	Se pueden identificar errores y solucionarlo durante el proceso
Herramientas de documentación	Historias de usuarios	Historias de usuarios	Tablero de tareas
Sprint o Iteraciones	Tareas concluidas no pueden realizar cambios futuros	Tareas concluidas pueden presentar cambios a futuro	Deberán cerrarse todas las tareas antes de empezar con la siguiente.
Equipo de desarrollo	Trabajo Individual	Trabajo en equipo (parejas)	Sin definir
Desarrollo de proyectos	Proyectos complejos y de inicio	Proyectos con requerimentación indefinida con cambios constantes	Proyectos de mantenimiento

Realizado por: Adriana Quille, 2019.

Con el objetivo de facilitar el desarrollo de la aplicación, se decidió utilizar la metodología ágil SCRUM debido a que facilita la administración y comunicación entre el equipo de trabajo y el cliente, esta metodología también permite el desarrollo de forma individual y realiza avances de funcionalidades por cada sprint.

1.16. SCRUM

SCRUM es una metodología ágil para trabajar en equipo ya sea por iteraciones o por Sprint, esta metodología se utiliza para planificar y controlar el proyecto de gran indecisión también se presenta cambios continuos durante el proceso de desarrollo.

La planificación se realiza por semanas y al final de cada sprint se realiza una revisión de validación con las tareas realizadas y los requisitos que vayan cambiando a corto plazo y en función de los resultados se priorizan y se planifican las siguientes actividades a desarrollar en las próximas semanas (López Gil, 2018, p.57).

1.16.1. Características

Según (López Gil, 2018, p.57), presenta las siguientes características:

- Desarrollo incremental en lugar de la clásica

- Planificación del desarrollo completo de un producto o servicio
- Equipo de trabajo auto organizados
- Se enfoca principalmente en el producto final, y en la calidad del mismo.

1.16.2. Roles de SCRUM

La metodología SCRUM posee con 3 roles fundamentales: Producto Owner, Scrum Master y equipo de desarrollo; además existe un agente externo conocido como cliente, siendo este el medio para iniciar con las necesidades del producto al Product Owner (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

A continuación, en la siguiente **Figura 7-1**, podemos observar los diferentes roles de la metodología SCRUM.

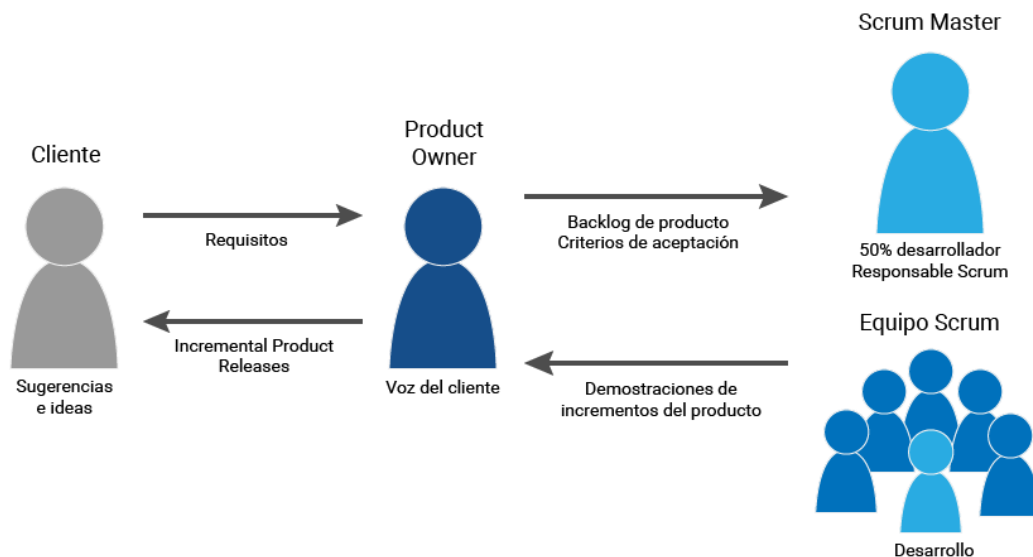


Figura 7-1: Roles de la metodología SCRUM

Fuente: (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

1. Product Owner

Conocido también como el propietario del producto; el mismo que se encargará del control y dirección del Product Backlog, como son las fechas de entregas que han sido establecidas por el equipo de desarrollo, también encargado de definir las diferentes funcionalidades del producto y toma de decisión del negocio, siendo la persona responsable para que el producto sea desarrollado de la mejor manera. Este Product Owner puede ser el mismo cliente del producto o una persona delegada por el cliente, el mismo que deberá asistir a todas las reuniones de planificación y revisión de los sprint y además poseer contacto directo con el equipo de desarrollo (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

2. Scrum máster

Persona encargada de hacer cumplir con todas las reglas que la metodología propone durante el proceso de desarrollo, guiando las reuniones, eliminando dificultades y dando solución inmediata, este rol es 50% de desarrollador y toma de decisiones rápidas, asegurando que el equipo de desarrollo continúe con el proceso sin ninguna dificultad, también es el representante del equipo de desarrollo frente al cliente (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

3. Equipo de desarrollo

Normalmente este equipo deberá estar formado por un número de personas, siendo 7 el número máximo y ± 2 personas como mínimo para el desarrollo de un producto, este equipo debe ser multifuncional, auto-organizado y mantener un contacto constante para hacer cumplir el objetivo del Sprint (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

Ciente

Persona beneficiaria del producto final, el mismo que durante el proceso de desarrollo puede aportar con ideas, sugerencias o dudas que se presenten por parte del equipo de desarrollo, su participación es muy fundamental (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

Artefactos

El principal artefacto es el propio producto, pero dentro de este contempla 3 artefactos principales que SCRUM utiliza durante el proceso de desarrollo, presentadas en la **Figura 8-1**:

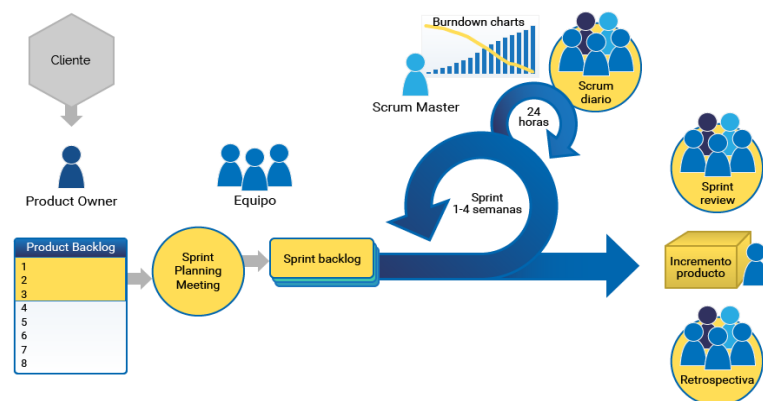


Figura 8-1: Artefactos de la metodología SCRUM

Fuente: (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

- **Product Backlog**

Contempla el listado de funcionalidades que deberán ser añadidas al producto, las mismas que deben ser priorizadas por el cliente, este producto backlog trata de las historias de usuario que se deberán describir por cada una de las funcionalidades que se vaya realizando durante el proceso de desarrollo del producto (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

- **Sprint Backlog**

El equipo de desarrollo es el encargado de crear el sprint de acuerdo a las funcionalidades priorizadas por el cliente, para cumplir con el objetivo de la funcionalidad descrita en el Producto Backlog (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

- **Burndown Chart**

Este artefacto trata de las entregas de cada uno de los sprint que se ha realizado con la finalidad de dar a conocer el cumplimiento de la planificación planteada con anterioridad (Desire 2017, <https://desire.webs.uvigo.es/contenidos/scrum/>).

1.17. Norma ISO/IEC 9126

La calidad de software define un conjunto de propiedades, características y entidades intelectuales (intangibles), como lo es el mejoramiento de procesos de software a través de técnicas y métodos estadísticos todos estos basándose en una arquitectura de software que le confiere su aptitud para satisfacer las necesidades explícitas o implícitas es decir que contribuye a mejorar la calidad de uso (Lemus Olalde, 2007, pp.9-10).

La Organización Internacional de Normalización creadora del estándar ISO/IEC 9126 en el año de 1992 con la finalidad de evaluar la calidad de software que permitirá examinar desde varios puntos característicos (Durán, 2015,p.28). El estándar ISO/IEC 9126 (Salazar, 2015, p.49) se encuentra formado por cuatro partes siendo estas modelo de calidad, métricas externas, métricas internas y métricas de calidad de uso.

Modelo de calidad. - Encargaba de describir las características de calidad que existen para un software, este modelo se divide en dos partes: calidad interna y externa, definiendo en la externa como calidad en uso (Salazar, 2015, <http://repositorio.espe.edu.ec/bitstream/21000/10128/1/T-ESPEL-MAS-0020.pdf>).

Calidad interna y externa. – ofrece seis características las mismas que cuentas con una subdivisión, mientras que la parte de calidad en uso cuenta con cuatro características pero no

posee subdivisiones (Salazar, 2015, <http://repositorio.espe.edu.ec/bitstream/21000/10128/1/T-ESPEL-MAS-0020.pdf>).

1.17.1. Características de la ISO/IEC 9126

Entre las características que la ISO/IEC 9126 presenta en la **Figura 9-1**, son las siguientes:

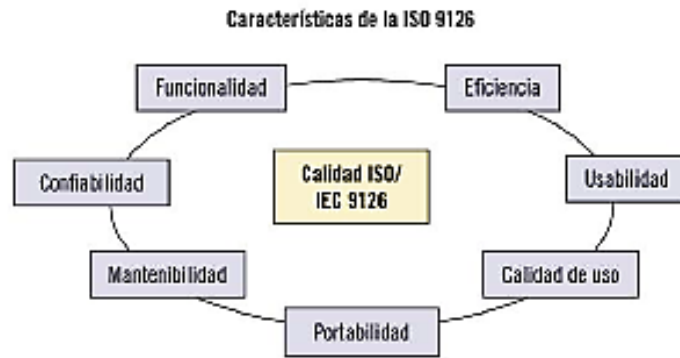
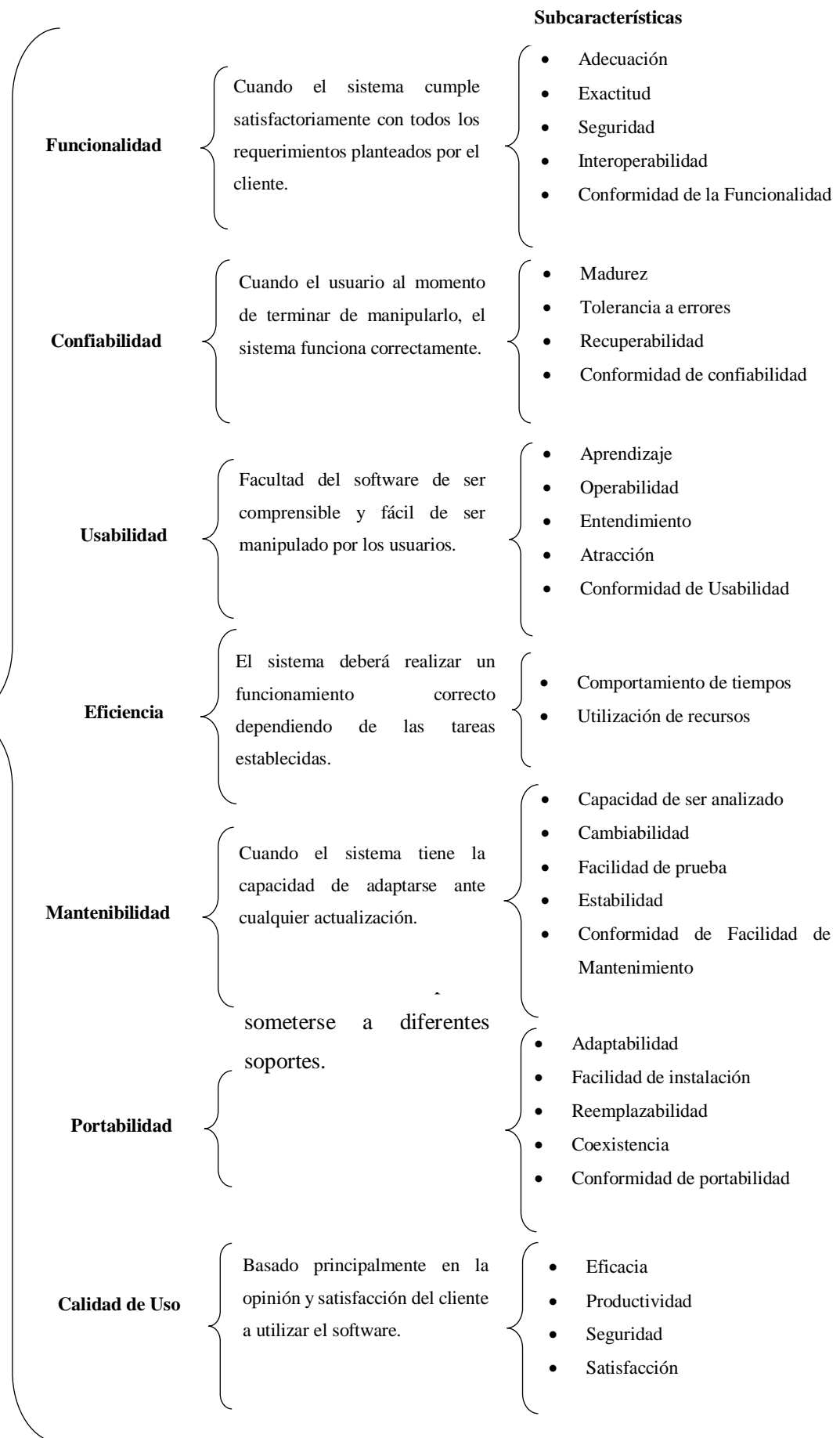


Figura 9-1: Características de la ISO/IEC 9126
Fuente: (Durán, 2015,p.29)

Características de la ISO/IEC 9126



Fuente: (Durán, 2015,pp.29-38).
Realizado por: Adriana Quille, 2019.

A continuación, en la **Figura 10-1** se observa un pequeño resumen de la ISO/IEC 9126.



Figura 10-1: Resumen de la ISO/IEC 9126.

Fuente: (Durán, 2015,p.39).

Todas las características propuestas por la norma mencionada establecen la calidad a las partes específicas del software; por lo que se pretende mejorar los tiempos de acceso haciendo uso del sistema web para el presente trabajo de titulación, el mismo que determinara la calidad del software en base a las características de eficiencia que esta norma ofrece.

1.17.2. Eficiencia

Esta métrica permite evaluar la relación entre atributos del nivel de desempeño de software y la cantidad de recursos necesarios, bajo condiciones establecidas como: el comportamiento de tiempos entre diversas actividades que se realizan frecuentemente en la fundación (Yarif Jadisha, 2010, <https://estandarsw.wordpress.com/category/iso/iso-9126/>).

La eficiencia del software es la forma del desempeño adecuado, de acuerdo con el número de recursos utilizados según las condiciones planteadas. Se debe tener en cuenta otros aspectos como la configuración de hardware, el sistema operativo, entre otros (Smartsys, 2011, <http://www.smartsys.com.ec/?p=391>).

Esta característica hace la relación al funcionamiento del software con la cantidad de recursos que se utiliza para su debido funcionamiento es decir el sistema operativo como también el hardware (Salazar, 2015, p.38).

- **Comportamiento de tiempos:** son aquellos tiempos de respuesta y procesamiento de un sistema cuando es ejecutada la respectiva funcionalidad en base a condiciones establecidas (Salazar, 2015, <http://repositorio.espe.edu.ec/bitstream/21000/10128/1/T-ESPEL-MAS-0020.pdf>).
- **Utilización de recursos:** encargada de la utilización de cada uno de los recursos que posee el software para su correcto funcionamiento bajo las condiciones ya determinadas (Durán, 2015,p.34).

Para el análisis del desarrollo del sistema web se utiliza la norma ISO/IEC 9126. La métrica por analizar en el presente trabajo de titulación es la eficiencia; en el cual se evaluará los comportamientos de tiempos de respuesta de los procesos de registrar emprendimientos, emprendedores y pedidos respectivamente donde, entre menor sea el tiempo de respuesta, mejor será para el cliente.

CAPÍTULO II

2. MARCO METODOLÓGICO

En el presente capítulo se indicará detalladamente como se llevó a cabo el desarrollo del sistema web para la Fundación “Emprender y No Dependier” de la ciudad de Riobamba provincia Chimborazo, la misma que fue desarrollada utilizando la metodología SCRUM, a continuación, se presentara las actividades que fueron realizadas para cumplir con las necesidades planteadas por el cliente.

2.1. Diseño de la Investigación

2.1.1. Tipo de investigación

En el presente trabajo de titulación se utilizará la investigación aplicada, con el propósito de aplicar los conocimientos teóricos y prácticos adquiridos a lo largo de la carrera académica, los mismos que ayudarán para el desarrollo del sistema web, satisfaciendo a los emprendedores con un producto funcional y eficiente.

2.1.2. Métodos de investigación

Los métodos de investigación que serán de gran ayuda para la realización del presente proyecto se describen a continuación:

- **Análisis:** se estudia todos los procesos de la gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”, con el objetivo de conocer y comprender como se lleva a cabo dichos procesos actualmente.
- **Síntesis:** este método se utiliza con la finalidad de recopilar información para desarrollo del marco teórico y poder explicar cada una de las herramientas que se utilizó durante el desarrollo del sistema web.
- **Inductivo:** primero se debe conocer el correcto funcionamiento del sistema, es decir, de lo particular a lo general; permitiendo llegar a las conclusiones del mismo, por tanto, esto se desarrolla en el marco de resultado discusión y análisis de resultados.
- **Deductivo:** este inicia de lo general a lo particular lo que conlleva a obtener recomendaciones a partir de las conclusiones obtenidas en el método inductivo.

2.1.3. Técnicas de investigación

Para la recopilación de la información necesaria para efectuar el desarrollo del sistema web, se establece el uso de las técnicas como es la entrevista, revisión de documentación de la fundación y la observación.

- Entrevista a la presidenta de la Fundación “Emprender y No Dependier”, así como también a la secretaria quien es encargada de recopilar la información de la fundación con el objetivo de conocer la función que cumple y los requerimientos del sistema.
- Revisión de documentación de la Fundación “Emprender y No Dependier” que es llevada de manera manual referente a la gestión y difusión de emprendimientos.
- Mediante la técnica de la observación en la Fundación “Emprender y No Dependier” se percibe el manejo de procesos de la gestión y difusión de los emprendimientos para analizar el cambio que existe en como llevaban esos procesos antes de desarrollar el sistema web.

2.2. Determinación de los procesos en la Fundación “Emprender y No Dependier”

2.2.1. Diagrama de procesos

Los procesos que la fundación posee actualmente fueron clasificados en función de los criterios: entrega y recepción de los emprendimientos y la gestión de pedidos; sin embargo, se clasificaron tomando en cuenta la priorización del negocio por cada historia de usuario que esté presente, dentro de los procesos tenemos directamente la gestión de emprendimientos para lo cual en la **Tabla 1-2**, se detalla los siguientes procesos:

Tabla 1-2: Procesos de la Fundación de manera no automatizada

N°	Actividades	Descripción de las actividades	Responsable	Resultado esperado
1	Entregar la carpeta con los documentos	El emprendedor entrega los documentos a la Fundación	Emprendedor	Entrega de documentos
2	Recibir la carpeta con los documentos	La fundación recibe los documentos del Emprendedor	Fundación EYND	Recibe los documentos
3	Analizar la carpeta con los documentos	La presidenta de la fundación analiza los documentos recibidos	Presidenta	Análisis de los documentos
4	Redactar el plan de Emprendimiento	La presidenta redacta el plan de trabajo para ser expuesto al emprendedor	Presidenta	Crea el plan del emprendimiento
5	Documentos incompletos u Erróneos	La presidenta analiza los documentos recibidos	Presidenta	Documentos completos

6	Recibir el plan de Emprendimiento	El emprendedor recibe el plan de emprendimiento emitido por la Fundación	Fundación	Conoce el plan del emprendimiento al que deberá adaptarse
7	Presentar el Emprendimiento a Ofertar	El emprendedor presenta el emprendimiento que desea ser ofertado en la Fundación	Emprendedor	Aprobación del emprendimiento para ser ofertado
8	Recibir el Emprendimiento a Ofertar	La fundación recibe el emprendimiento a ofertar	Fundación	Análisis del emprendimiento a ofertar
9	Consultar emprendimientos	Los usuarios pueden visualizar los diferentes emprendimientos que la Fundación se encuentra ofertando	Usuarios	Comprobación que exista emprendimientos en exhibición
10	Seleccionar emprendimientos	Los usuarios pueden seleccionar los diferentes emprendimientos	Usuarios	Escoge los emprendimientos de acuerdo a su criterio
11	Crear pedido	Los usuarios realizan una llamada telefónica a la Fundación	Usuarios	Recepción de la llamada del / los usuarios
12	Registrar pedido	La persona encargada de recibir la llamada realiza el registro del pedido	Secretaria	Análisis y registro del pedido
13	Calcular el precio total	Realiza una suma de todos los emprendimientos de acuerdo al pedido registrado	Secretaria	Obtiene un valor calculado para verificar si el pedido se concluye
14	Enviar información del pedido	La fundación comunicara al emprendedor sobre el pedido registrado	Secretaria	Confirmación del envío del registro de pedido
15	Recibir información del pedido	El emprendedor recibe el registro del pedido	Emprendedor	Recepción del registro del pedido
16	Enviar confirmación del pedido	El emprendedor confirma el pedido	Emprendedor	Confirmación del pedido registrado
17	Procesar pedido	El emprendedor empieza a trabajar en el pedido registrado	Emprendedor	Confecciona el emprendimiento a entregar
18	Recibir confirmación del pedido	La fundación se comunica con el emprendedor para confirmar detalles de pedido	Secretaria	Detalles de confirmación del pedido
19	Enviar pedido	El emprendedor realiza el envío del pedido al usuario	Emprendedor	Envío del pedido
20	Recibir pedido	El usuario recibe el pedido que lo realizo con anterioridad	Usuario	Recibió conforme el pedido que lo realizo

Realizado por: Adriana Quille, 2019.

A continuación, en la **Figura 1-2, Figura 2-2**, se presenta el respectivo diagrama de procesos el mismo que describe paso a paso como se realizan los procesos actualmente en la fundación dando a conocer la actividad y descripción con la que cuenta

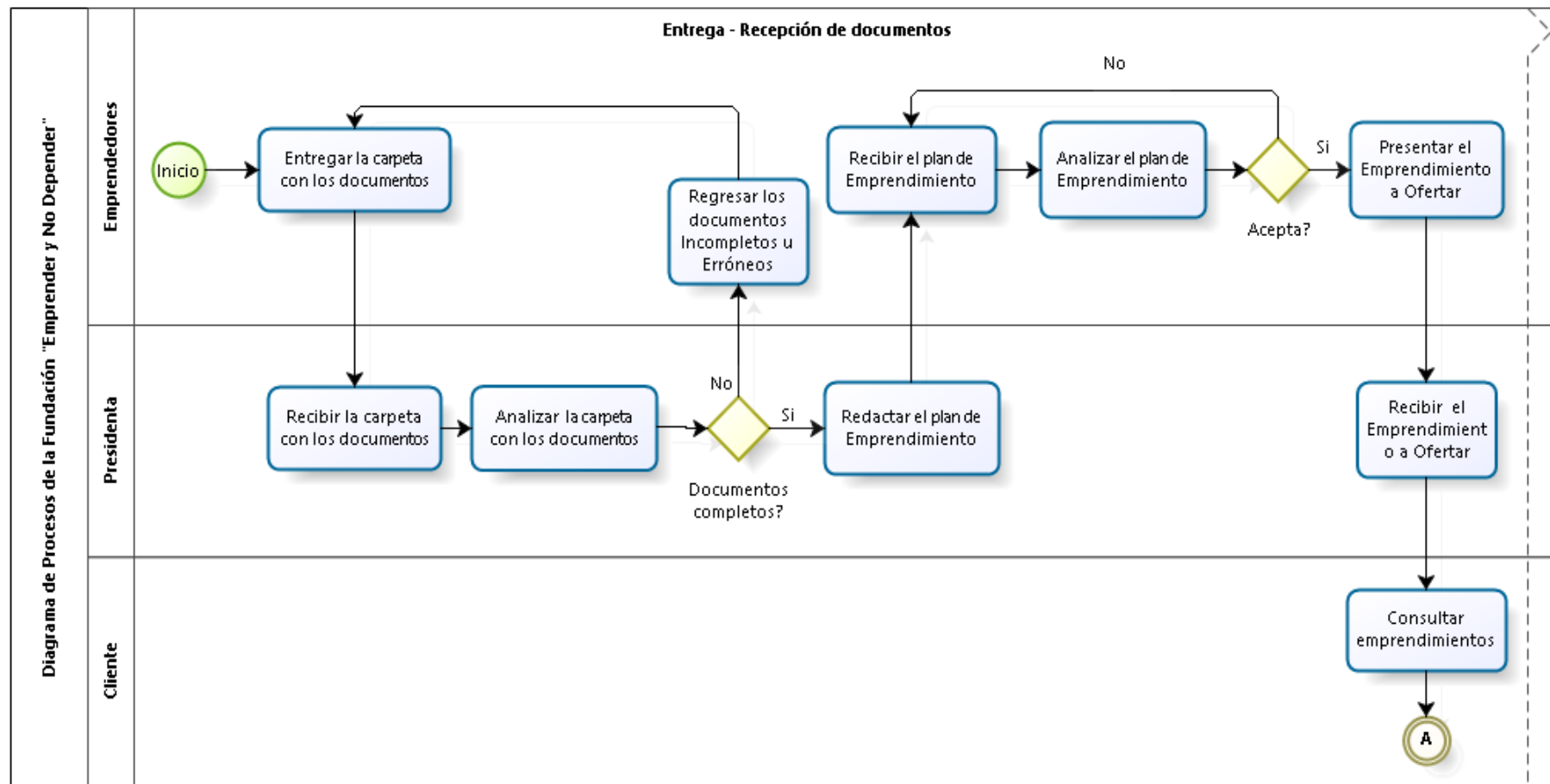


Figura 1-2: Diagrama de procesos parte 1.

Realizado por: Adriana Quille, 2019

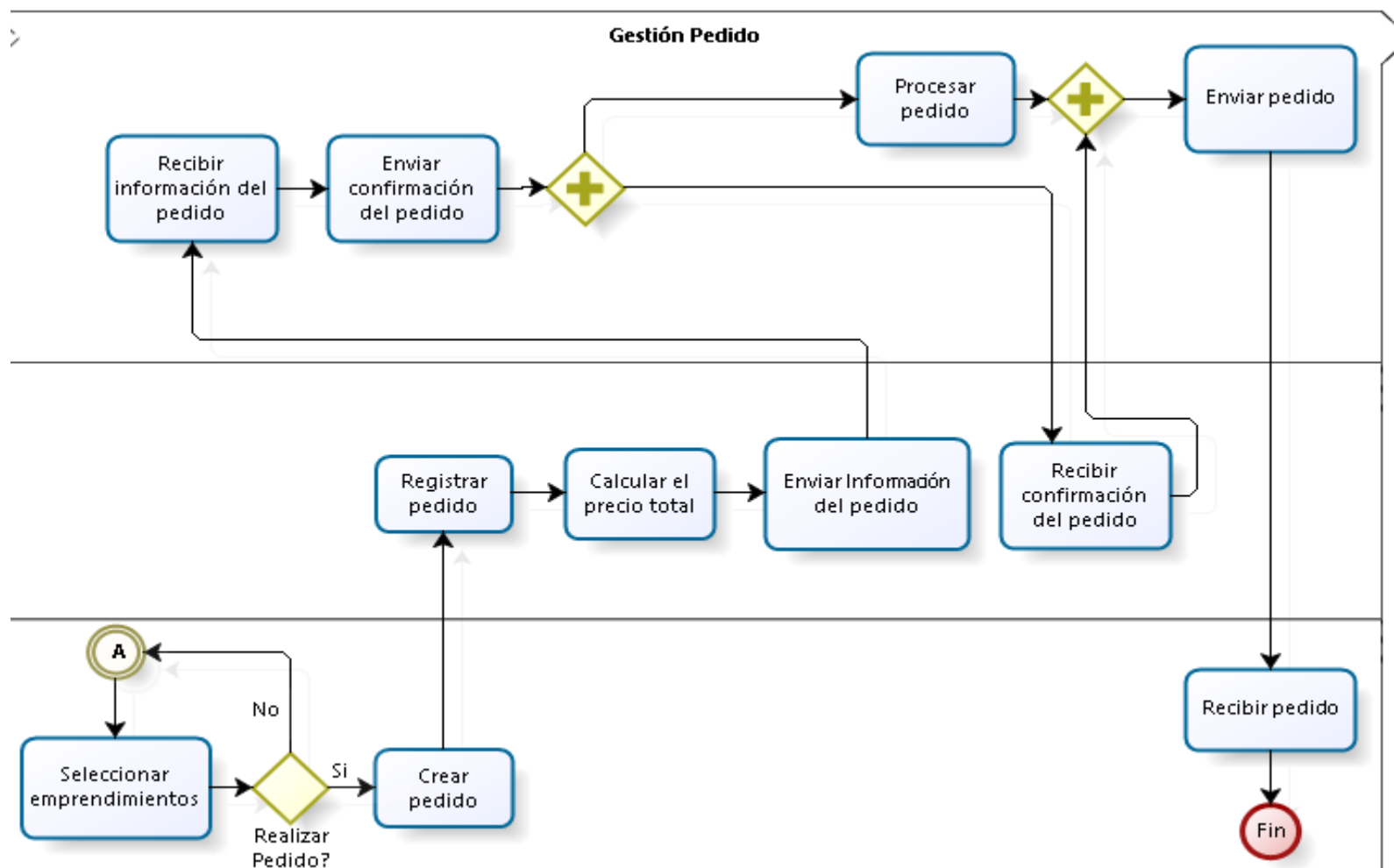


Figura 2-2: Diagrama de procesos parte 2.
 Realizado por: Adriana Quille, 2019

2.2.2. Diagrama de procesos sistematizado

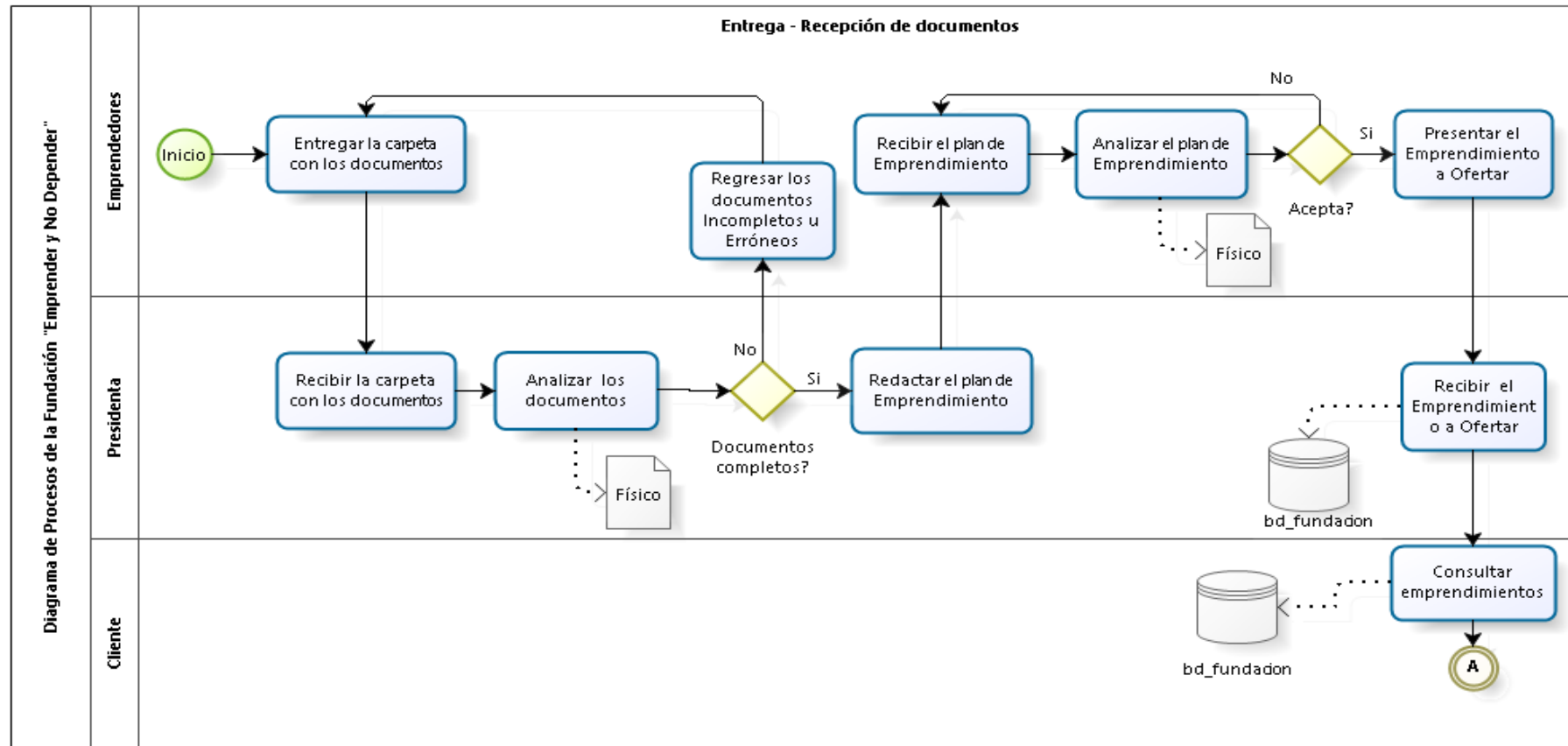


Figura 3-2: Diagrama de procesos sistematizado parte 1.
 Realizado por: Adriana Quille, 2019

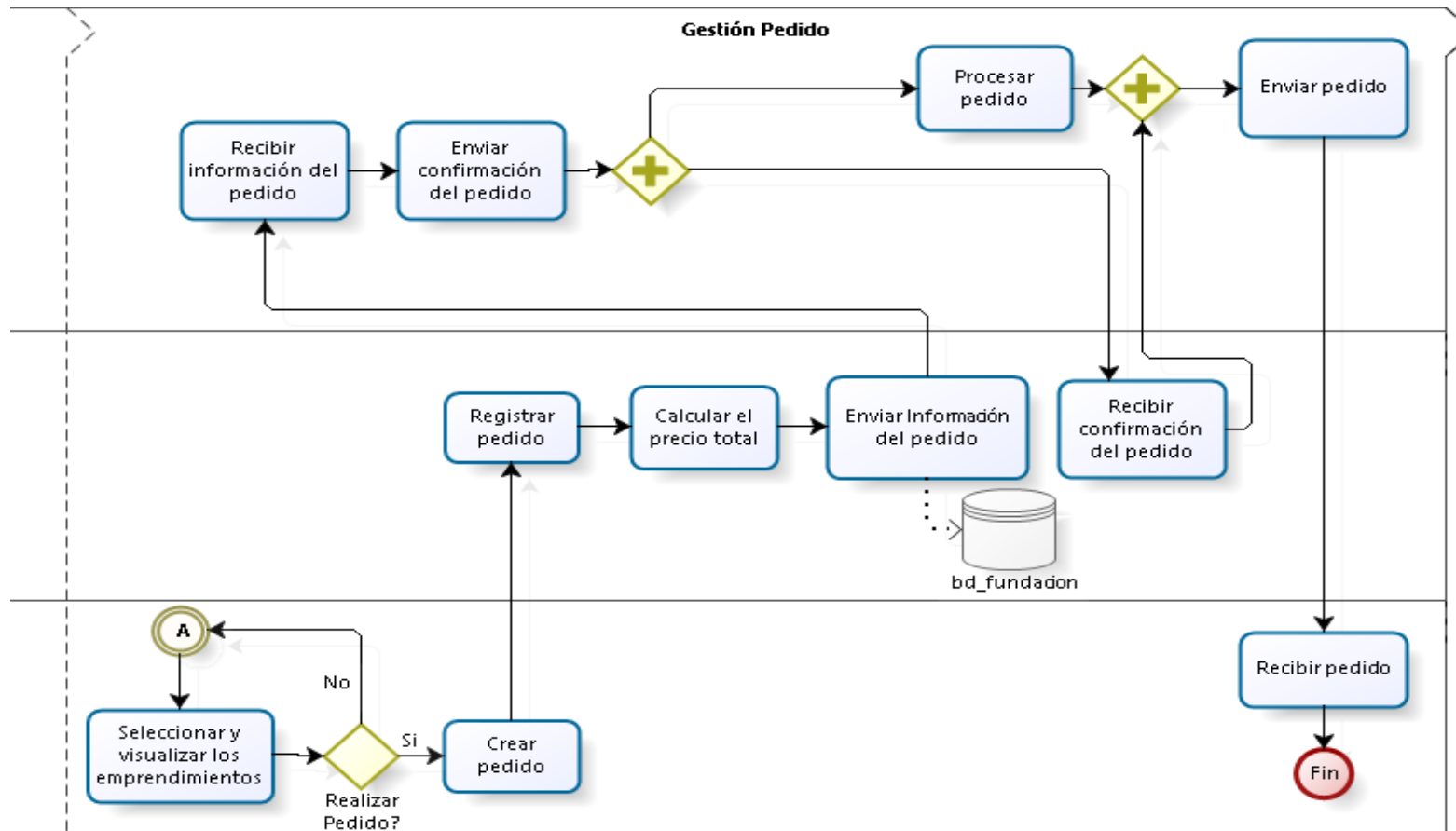


Figura 4-2: Diagrama de procesos sistematizado parte 2.
 Realizado por: Adriana Quille, 2019

En las **Figura 3-2** y **Figura 4-2**, se presenta el diagrama de procesos sistematizado el mismo que describe paso a paso como se encuentran desarrollados los procesos en el sistema web.

2.3. Desarrollo de la metodología ágil SCRUM

2.3.1. Fase de planificación

En esta fase de planificación se realizará un estimado de las fechas en las que se llevará a cabo las actividades establecidas para el desarrollo de este trabajo de titulación, también la recolección de información, selección de herramientas, el desarrollo del sistema, las pruebas de implementación del sistema y la respectiva capacitación para los usuarios.

2.3.1.1. Personas involucradas

El equipo SCRUM encargado del desarrollo del sistema web está conformado por 3 personas los cuales se detallan en la siguiente **Tabla 2-2** con su respectivo rol.

Tabla 2-2: Roles y personas

Persona	Rol	Contacto	Institución
Ing. Eduardo Villa	SCRUM Master	060182603-5 evilla@epoch.edu.ec	ESPOCH – FIE
Ing. María Murillo	Product Owner	gruaem@gmail.com	Fundación Emprender y No Dependier
Adriana Quille	Development Team	020234482-6 adriana.quille@epoch.edu.ec	ESPOCH – FIE

Realizado por: Adriana Quille, 2019.

2.3.1.2. Tipos y roles de usuario

A continuación, se describe los tres tipos de roles para los usuarios quienes poseen diversas funcionalidades en el sistema, las cuales se describen en la siguiente **Tabla 3-2**.

Tabla 3-2: Tipos de usuario y roles.

Tipo De Usuario	Descripción	Rol
Administrador	Encargado de manejar y administrar el sistema	Ing. María Murillo
Emprendedor	Encargado de publicar sus emprendimientos	Emprendedores
Cientes	Pueden observar la información del sistema, es decir servicios y productos que el sistema ofrece	Cientes

Realizado por: Adriana Quille, 2019.

2.3.1.3. Planificación

La metodología ágil empleada para el desarrollo de este proyecto de titulación es SCRUM, siendo esta quien nos obliga a cumplir con cada uno de los requisitos ya planteados, SCRUM permite realizar un listado de requisitos tanto técnicos como de usuario cada uno de estas enlazadas con su respectivo Sprint de acuerdo a la prioridad que se le ha otorgada a cada uno de los requisitos que posee el sistema, por tanto se realiza la respectiva estimación haciendo uso del método de la talla de la camiseta o t-shirt que permite otorgar estimaciones S, M, XM Y L como se presenta en la **Tabla 4-2**.

Tabla 4-2: Estimaciones para el sistema.

Talla	Puntos Estimados	Horas de trabajo
XS	8	8
S	16	16
XM	20	20
M	40	40
L	80	80

Realizado por: Adriana Quille, 2019.

Para cada tarea se deberá estimar, nótese que un punto estimado es igual a una hora de trabajo lo que implica que un día de trabajo es equivalente a 8 horas realizado por una sola persona por consecuencia una semana de trabajo es de 5 días lo que equivale a 40 puntos estimados, dos semanas de trabajo con 10 días equivalente a 80 puntos estimados siendo el valor máximo y respectivo que debe poseer cada Sprint, el mismo que se lo realiza cada 2 semanas respectivamente.

2.3.1.4. Producto Backlog

En el producto backlog se detalla los requisitos expresados por el cliente durante las reuniones con la Ing. María Murillo (Product Owner) y los miembros del desarrollo (Development Team) **Anexo A**, todos estos requisitos se lo ha clasificado en historias técnicas (HT) e historias de usuarios (HU) con los que aseguramos el correcto desarrollo de la aplicación, además cada HU y cada una de las HT cuentan con su nivel de prioridad las cuales están dadas en un rango del 1 al 10 en donde 1 es la prioridad baja y 10 es la prioridad alta.

A continuación, se visualiza el Product Backlog en la **Tabla 5-2**.

Tabla 5-2: Producto backlog del sistema web.

ID	HISTORIA	Prioridad	Estimación
HT-01	Definir el estándar de codificación para el desarrollo del sistema FEYND.	ALTA	20
HT-02	Definir la arquitectura del sistema FEYND	ALTA	20
HT-03	Definir el estándar de interfaces de usuario para el desarrollo del sistema.	ALTA	40
HT-04	Instalar los framework que se utilizara para el desarrollo del sistema.	ALTA	20
HT-05	Diseño e Implementación de la base de datos	ALTA	40
HT-06	Investigación del Framework Symfony y Bootstrap	MUY ALTA	80
HT-07	Capacitación a los usuarios	BAJA	8
HT-08	Manual de usuario.	ALTA	40
HU-01	Como emprendedor, requiero registrar los datos de los productos con la finalidad de que se almacene en la base de datos	ALTA	20
HU-02	Como emprendedor, requiero actualizar los datos del producto con la finalidad de agregar o eliminar características que poseen	ALTA	16
HU-03	Como emprendedor, requiero buscar productos con la finalidad de realizar acciones que sean necesarias	MEDIA	16
HU-04	Como emprendedor, deseo listar productos con la finalidad de conocer todos los productos existentes	BAJA	8
HU-05	Como emprendedor, deseo eliminar productos con la finalidad de quitar del listado de productos existentes	BAJA	8
HU_06	Como administrador requiero registrar los datos de las categorías con la finalidad de que se almacene en la base de datos	ALTA	20
HU_07	Como administrador requiero actualizar los datos de categorías con la finalidad de agregar o eliminar características que poseen	ALTA	20
HU_08	Como administrador requiero buscar categorías con la finalidad de realizar acciones que sean necesarias	MEDIA	16
HU_09	Como administrador deseo listar categorías con la finalidad de conocer todos los productos existentes	MEDIA	16
HU_10	Como administrador deseo eliminar categorías con la finalidad de quitar del listado de categorías existentes	MEDIA	16
HU-11	Como administrador, requiero registrar los datos con la finalidad de que se almacene en la base de datos	MEDIA	16
HU-12	Como cliente, requiero registrar los datos con la finalidad de que se almacene en la base de datos.	MEDIA	16
HU-13	Como administrador, requiero actualizar datos con la finalidad de agregar o eliminar características que poseen	MEDIA	16
HU-14	Como administrador, deseo buscar usuario con la finalidad de realizar acciones que sean necesarias	MEDIA	16
HU-15	Como administrador, requiero listar usuarios con la finalidad de conocer todos los usuarios existentes	MEDIA	16
HU-16	Como administrador, deseo eliminar usuarios con la finalidad de quitar del listado de usuarios existentes.	BAJA	8
HU-17	Como cliente, requiero realizar un pedido con la finalidad de poder vender los emprendimientos a los clientes lo más pronto posible	ALTA	20

HU-18	Como cliente, necesito modificar el pedido que aún no han sido autorizado para su respectiva entrega.	ALTA	20
HU-19	Como cliente, deseo actualizar los datos que poseen en el sistema para agregar o modificar futura información.	MEDIA	16
HU-20	Como administrador, requiero que se genere las notificaciones para que los emprendedores conozcan del pedido realizado	ALTA	20
HU-21	Como cliente, requiero ingresar una calificación con la finalidad de dar a conocer el producto garantizado.	ALTA	20
HU-22	Como administrador, requiero poder modificar los pedidos realizados por los clientes para agregar o modificar futura información.	ALTA	20
HU-23	Como administrador, requiero ingresar datos del emprendedor con la finalidad de que se almacene en la base de datos	ALTA	20
HU-24	Como emprendedor, deseo actualizar la información con la finalidad de agregar o eliminar características que posee	MEDIA	16
HU-25	Como administrador, requiero buscar emprendedor con la finalidad de realizar acciones que sean necesarias	MEDIA	16
HU-26	Como administrador, requiero listar emprendedores con la finalidad de conocer todos los emprendedores existentes	MEDIA	16
HU-27	Como administrador requiero eliminar (cambiar de estado) emprendedor con la finalidad de quitar del listado de emprendedores existentes	BAJA	8
HU-28	Como administrador, requiero registrar la información de los usuarios del sistema para gestionarla.	ALTA	20
HU-29	Como administrador, requiero crear los diferentes usuarios (roles), para otorgar los respectivos permisos para acceder al sistema.	ALTA	20
HU-30	Como administrador, necesito modificar datos de los usuarios con la finalidad de agregar o modificar futura información.	ALTA	20
HU-31	Como administrador, requiero buscar cliente con la finalidad de realizar acciones que sean necesarias	MEDIA	16
HU-32	Como administrador, deseo listar clientes con la finalidad de conocer todos los clientes existentes	MEDIA	16
HU-33	Como administrador, deseo ingresar la información general de la fundación con la finalidad de que la información sea conocida por nuestros clientes.	ALTA	20
HU-34	Como administrador, deseo actualizar la información de la fundación con la finalidad de agregar o modificar futura información	MEDIA	16
HU-35	Como desarrollador, requiero desarrollar el autenticar de usuarios para el Sistema FEYND.	ALTA	20
HU-36	Como administrador, necesito buscar y cambiar de estado a los usuarios del sistema.	MEDIA	16
HU-37	Como administrador, requiero listar los emprendimientos por categoría con la finalidad de conocer todos estos.	MEDIA	16
HU-38	Como administrador, requiero listar a los emprendedores por categoría con la finalidad de conocer el número de emprendedores existentes.	MEDIA	16
HU-39	Como administrador, requiero listar a los emprendedores que se encuentran o no calificados.	ALTA	20

HU-40	Como administrador, requiero listar los pedidos de forma genera.	ALTA	20
HU-41	Como cliente, deseo comunicarme con el respectivo administrador con la finalidad de acordar el producto para el respectivo pedido.	ALTA	20

Realizado por: Adriana Quille, 2019.

2.3.1.5. Sprint backlog

En cada uno de los Sprint se describen el desarrollo de cada una de las actividades del sistema, cada sprint cuenta con las diferentes historias de usuario e historias técnicas, como también la duración de una semana en la cual se desarrollan las funcionalidades del sistema por el personal de equipo de desarrollo. El Sprint backlog se puede observar en la **Tabla 6-2**.

Tabla 6-2: Sprint backlog del sistema web.

Sprint	ID	Prioridad	Estimación	Fecha Inicio	Fecha Fin	N° Hora
1	HT-02	ALTA	20	22/10/2018	24/10/2018	80
	HT-03	ALTA	40	24/10/2018	31/10/2018	
	HT-04	ALTA	20	31/10/2018	2/11/2018	
2	HT-06	MUY ALTA	80	5/11/2018	16/11/2018	80
3	HT-05	ALTA	40	19/11/2018	23/11/2018	80
	HT-01	ALTA	20	26/11/2018	28/11/2018	
	HU-01	ALTA	20	28/11/2018	30/11/2018	
4	HU-02	ALTA	16	3/12/2018	4/12/2018	80
	HU-03	MEDIA	16	5/12/2018	6/12/2018	
	HU-05	BAJA	8	7/12/2018	7/12/2018	
	HU-06	ALTA	20	10/12/2018	12/12/2018	
	HU-07	ALTA	20	12/12/2018	14/12/2018	
5	HU_08	MEDIA	16	17/12/2018	18/12/2018	80
	HU_09	MEDIA	16	19/12/2018	20/12/2018	
	HU_10	MEDIA	16	21/12/2018	24/12/2018	
	HU_11	MEDIA	16	25/12/2018	26/12/2018	
	HU_12	MEDIA	16	27/12/2018	28/12/2018	
6	HU-04	BAJA	8	31/12/2018	31/12/2018	80
	HU-13	MEDIA	16	1/1/2019	2/1/2019	

	HU-14	MEDIA	16	3/1/2019	4/1/2019	
	HU-17	ALTA	20	7/1/2019	9/1/2019	
	HU-18	ALTA	20	9/1/2018	11/1/2019	
7	HU-15	MEDIA	16	14/1/2019	15/1/2019	80
	HU-16	BAJA	8	16/1/2019	16/1/2019	
	HU-19	MEDIA	16	17/1/2019	18/1/2019	
	HU-20	ALTA	20	21/1/2019	23/1/2019	
	HU-21	ALTA	20	23/1/2019	25/1/2019	
8	HU-22	ALTA	20	28/1/2019	30/1/2019	80
	HU-23	ALTA	20	30/1/2018	1/2/2019	
	HU-24	MEDIA	16	4/2/2019	5/2/2019	
	HU-26	MEDIA	16	6/2/2019	7/2/2019	
	HU-27	BAJA	8	8/2/2019	8/2/2019	
9	HU-28	ALTA	20	11/2/2019	13/2/2019	80
	HU-29	ALTA	20	13/2/2019	15/2/2019	
	HU-30	ALTA	20	18/2/2019	20/2/2019	
	HU-33	ALTA	20	20/2/2019	22/2/2019	
10	HU-31	MEDIA	16	25/2/2019	26/2/2019	80
	HU-32	MEDIA	16	27/2/2019	28/2/2019	
	HU-34	MEDIA	16	1/3/2019	4/3/2019	
	HU-36	MEDIA	16	5/3/2019	6/3/2019	
	HU-37	MEDIA	16	7/3/2019	8/3/2019	

11	HU-25	MEDIA	16	11/3/2019	12/3/2019	80
	HU-38	MEDIA	16	13/3/2019	14/3/2019	
	HU-39	ALTA	20	15/3/2019	19/3/2019	
	HU-40	ALTA	20	19/3/2019	21/3/2019	
	HT-07	BAJA	8	22/3/2019	22/3/2019	
12	HU-35	ALTA	20	25/3/2019	27/3/2019	80
	HU-41	ALTA	20	27/3/2019	29/3/2019	
	HT-08	Alta	40	1/4/2019	5/4/2019	

Realizado por: Adriana Quille, 2019.

Se obtuvo un total de 12 sprint con fecha de inicio lunes 22 de octubre de 2018 y fecha de finalización viernes 5 de abril del 2019 con un total de 960 puntos estimados.

2.3.1.6. Reuniones SCRUM

Utilizando la metodología se pueden fijar varias reuniones entre los miembros del equipo, de manera que se podrá especificar o aclarar las ideas consistentes de las diversas actividades o funcionalidades que se requiera para así satisfacer al cliente el mismo que también forma parte de las reuniones de manera que pueda exponer sus dudas o ideas más claras para el desarrollador.

En el proceso del desarrollo de este trabajo de titulación se realizó reuniones de inicio y fin de sprint, en la de inicio la desarrolladora se reunió para expresar cuáles serán las funcionalidades que se iban a realizar en dicho sprint que se aproximaba a desarrollar, por otra parte, la reunión de fin de sprint se revisó cada una de las actividades o funcionalidades que estaban planificadas en el sprint las mismas que ya se encuentran culminadas.

También por otra parte se realizaron reuniones semanales con el equipo de desarrollo director del proyecto y desarrolladora, para informar de algún inconveniente presentado y dar solución a la misma o para aclarar dudas.

2.3.2. Fase de desarrollo

En esta fase de desarrollo se presenta cada una de las diversas actividades que se llevaron a cabo durante el proceso de desarrollo del sistema web para la fundación “Emprender y No Dependier”.

2.3.2.1. Gestión de riesgos

Los riesgos son considerados como amenazas que pueden producirse en cualquier instante durante el proceso de desarrollo del sistema, el mismo que tiene repercusiones en la elaboración del proyecto llegando hasta a consecuencias extremas como puede ser el fracaso del proyecto.

Por tanto, es necesario tener una preparación para cuando se presenten cualquier tipo de riesgo o en el mejor de los casos se debe realizar trabajos preventivos para evitar estos riesgos en lo posible que no se presenten y así evitar problemas durante el proceso de desarrollo del proyecto, es por ello que se debe contar con un control de riesgos y para ellos realizar el respectivo análisis **ANEXO B.**

Como información en el análisis de riesgos se obtuvo un total de ocho riesgos que pueden presentarse en el proceso de desarrollo del proyecto los cuales se les clasifico según tres tipos, siendo estos riesgos del proyecto, riesgo técnico y riesgo del negocio, a estos tipos de riesgos se lo realizado la priorización basándose en el porcentaje de que suceda el riesgo, el impacto que tendrá en el proyecto y la exposición del mismo, dado esto se realizó las hojas de gestión del riesgo, en estas se presentan las acciones que se deberán realizar poder prevenir la aparición de estas amenazas.

A continuación, en la **Tabla 7-2** se presenta los riesgos que han sido identificados para este trabajo de titulación:

Tabla 7-2: Riesgos identificados en el proyecto.

ID	DESCRIPCIÓN	TIPO DE RIESGO	CONSECUENCIA
R1	Cambio constante de requerimientos.	Proyecto	<ul style="list-style-type: none">• Replanificación. del proyecto• Aumento de costos.• Retraso del proyecto.
R2	Falta de compromiso por parte del usuario.	Proyecto	<ul style="list-style-type: none">• Retraso en la entrega del proyecto• Información inconsistente en el desarrollo.• Mal diseño del sistema.

			<ul style="list-style-type: none"> • Funcionalidades del sistema sin aprobación del usuario.
R3	Pérdida de los equipos de trabajo.	Proyecto	<ul style="list-style-type: none"> • Retraso en el proyecto • Pérdidas económicas • Posible fracaso del proyecto.
R4	Erróneo diseño de la base de datos	Técnico	<ul style="list-style-type: none"> • Retraso en el proyecto • Rediseño de la base de datos
R5	El equipo no está familiarizado con la herramienta de desarrollo	Técnico	<ul style="list-style-type: none"> • Retraso en el desarrollo del proyecto. • Tiempo perdido
R6	Falta de acceso a internet	Técnico	<ul style="list-style-type: none"> • Pérdida de tiempo • Retraso del proyecto en caso de necesitar información para continuar en el desarrollo.
R7	La fundación decide cancelar las actividades	Negocio	<ul style="list-style-type: none"> • Fracaso del proyecto
R8	Falta de compromiso por parte del jefe de la Fundación	Negocio	<ul style="list-style-type: none"> • Información inconsistente para el desarrollo. • Mal diseño del sistema. • No satisface los requerimientos del usuario. • No cumple con la funcionalidad del sistema. • No se puede concluir el proyecto.

Realizado por: Adriana Quille, 2019.

2.3.2.2. Arquitectura del sistema

La arquitectura del sistema con el que fue desarrollado es de gran importancia para el desarrollo del sistema web permitiendo conocer cómo se encuentra estructurado el sistema y la forma con la que trabajan entre sí, a continuación, en la **Figura 5-2**, se presenta como ha sido implementado.

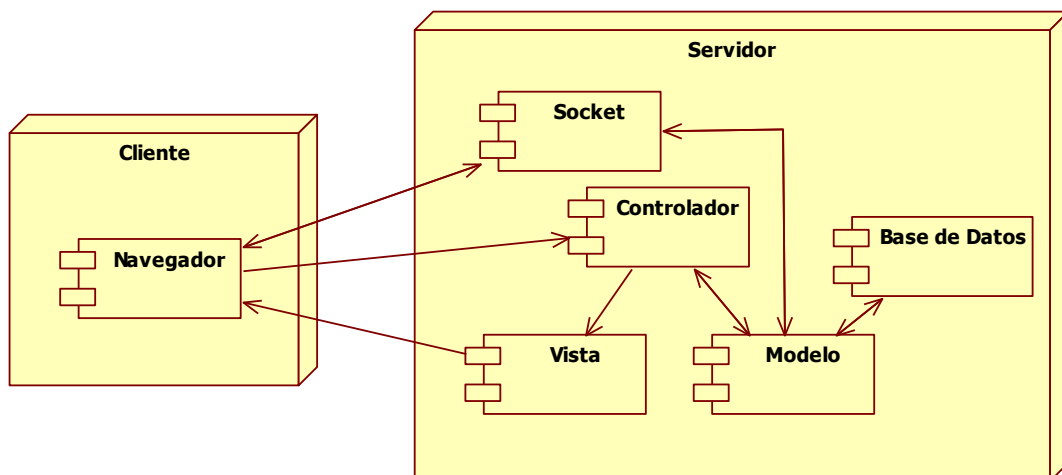


Figura 5-2: Arquitectura del sistema web.

Realizado por: Adriana Quille, 2019.

La arquitectura implementada en este software es Cliente/Servidor en la que se utilizó el patrón de diseño MVC, el mismo que está compuesto por la vista lugar donde reposa las interfaces de los diferentes usuarios, modelo paquete en donde se encuentra todo el acceso de datos, permitiendo consumir funciones para tener acceso a la base de datos, y finalmente el controlador que es el encargado de ejecutar las reglas del negocio y el servidor de base de datos se encuentra en el motor de base de datos de MySQL, lugar donde se almacenara y se realizara las diversas operaciones de cada uno de los requerimientos indicados para el correcto funcionamiento del sistema.

2.3.2.3. Diseño del sistema web

A continuación, en la **Figura 6-1**, puede observar los diversos módulos que fueron desarrollados para el sistema web.

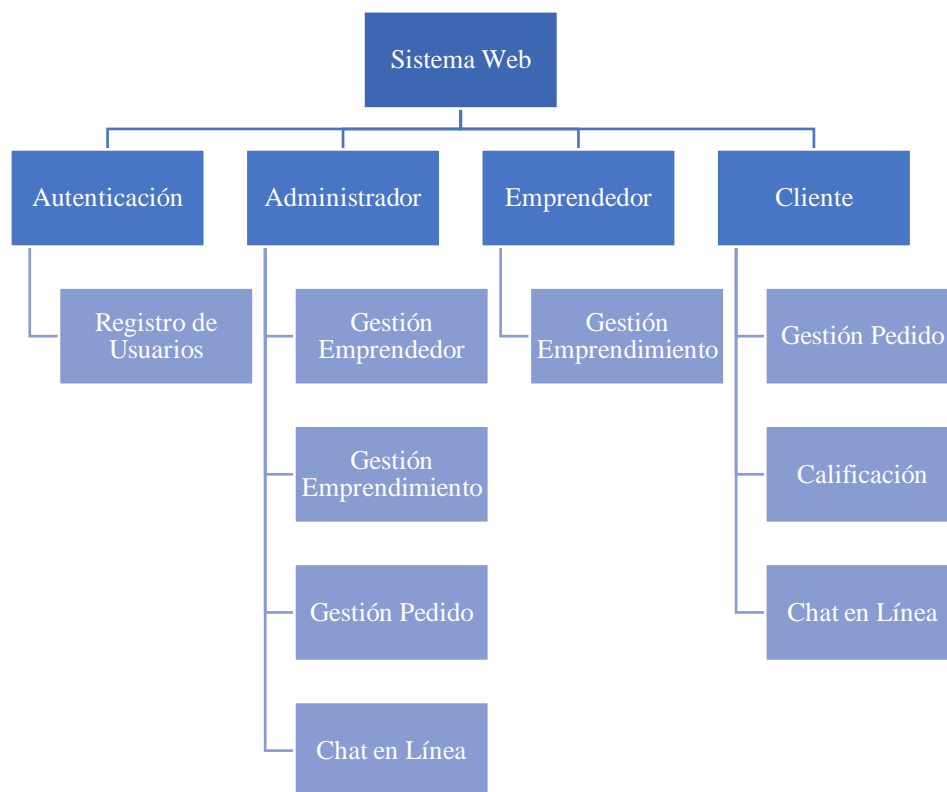


Figura 6-1: Sistema Automatizado.

Realizado por: Adriana Quille, 2019.

- **Autenticar:** este módulo permitirá realizar la autenticación de los usuarios (administrador, emprendedor, cliente).
- **Administrador:** contiene los módulos de: *gestión de emprendedor* en el cual se puede realizar el ingreso, modificación, eliminación y listado de emprendedores, *gestión de emprendimiento* tiene las opciones de ingresar, modificar, eliminar y listar emprendimientos,

gestión de pedido permitirá el ingreso, modificación, eliminación y listado de pedidos, y el *chat en línea* que permite la comunicación bidireccional.

- **Emprendedor:** posee el módulo de: *gestión de emprendimiento* que permitirá realizar el ingreso y modificación de los emprendimientos.
- **Cliente:** contiene los módulos de: *gestión de pedido* en el cual se puede realizar el ingreso y modificación de los pedidos, además podrá efectuar la *calificación* de los emprendimientos y llevar acabo el *chat en línea*.

2.3.2.4. Estándar de codificación

Para el desarrollo del sistema web se empleó el estándar de codificación CamelCase con el objetivo de que el código que se genera para la aplicación tenga características como uniformidad, también que sea legible y sobre todo que exista comprensión al momento de realizar la integridad de datos generado, de esta manera en este estándar se identifica cosas generales como la identificación de variables, constantes, clases, atributos y métodos, como se muestra en la **Tabla 8-2**.

Tabla 8-2: Estándar de codificación.

	Tipo	Estándar	Ejemplo
Base de datos	Tablas	CamelCase	NombreTabla
	Primary key	lowerCase	Id
	Atributos	mixedCase	nombreAtributo
Código	Bundle	CamelCase	BundleNombre
	Clase	CamelCase	NombreClase
	Método	mixedCase	nombreMetodo
	Variable	Lowercase	Variable
	Constante	UPPERCASE	NOMBRECONSTANTE

Realizado por: Adriana Quille, 2019.

Como puede observar en la tabla expuesta se define el respectivo estándar tanto para la base de datos como para el código de tal manera que esta se encuentra desarrollada en el sistema respetivamente.

2.3.2.5. Estándar de Interfaces

La estandarización de interfaces se lo realizó para mantener una misma analogía en las pantallas del sistema, misma que se determinó mediante las reuniones realizadas con el cliente.

En la **Figura 7-2**, se muestra la pantalla del sistema con los diferentes elementos que forman parte del rol administrativo, el mismo que cuenta con un menú superior (zona de menú) con las diversas actividades como es Productos, Carrito de compras, registro y Login.



Figura 7-2: Página de Inicio.
Realizado por: Adriana Quille, 2019.

Luego de generar todos los bosquejos de las interfaces de usuario, a continuación, se mostrará el resultado final de la interfaz realizada y cumplimiento de los requisitos que el cliente lo sugirió, así como en la **Figura 8-2**, puede observar el agregar emprendimiento donde deberá llenar cada uno de los campos como son *Código del emprendimiento*, *Nombre*, *Categoría*, *Precio*, *Descripción*, *Unidades disponibles*, *Nombre del Emprendedor* y la respectiva *Imagen del emprendimiento* que se está registrando; además se visualiza un botón Añadir el mismo que al dar clic sobre él, el emprendimiento ingresado será añadido a la Fundación.

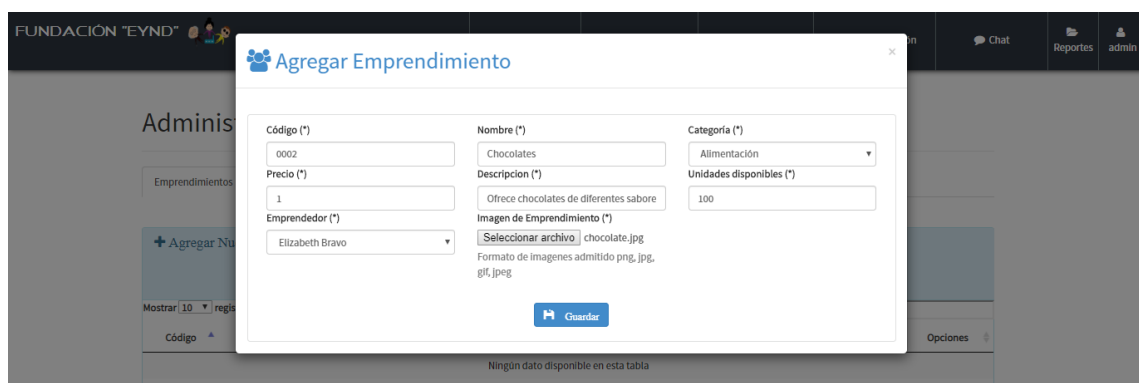


Figura 8-2: Modal del ingreso de emprendimientos
Realizado por: Adriana Quille, 2019.

En la **Figura 9-2**, puede visualizar una ventana modal que se ubica en el centro de la ventana, la misma que informa que el *Emprendimiento ha sido agregado con éxito* a la Fundación.

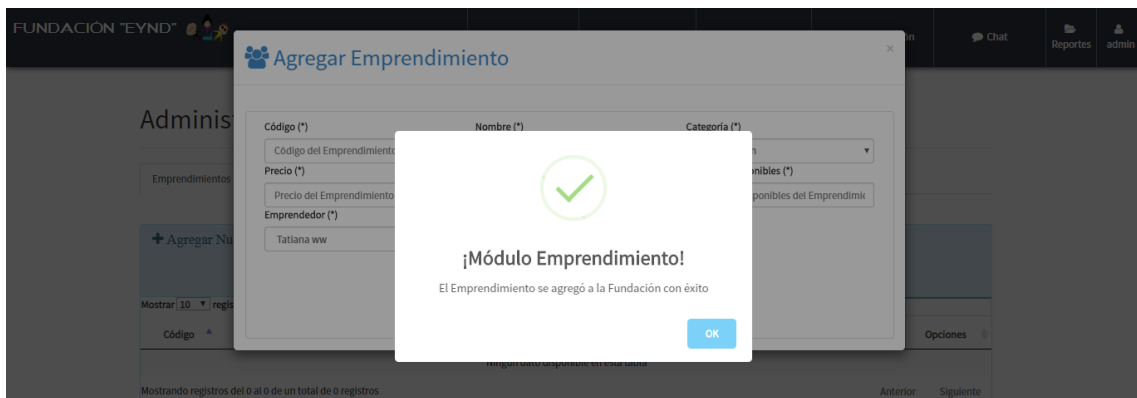


Figura 9-2: Interfaz del emprendimiento agregado.

Realizado por: Adriana Quille, 2019.

Cabe mencionar que todas las interfaces del sistema web se encuentran elaboradas de la misma manera que se describió en las figuras antes expuestas, para más detalles acerca del estándar de interfaces puede dirigirse al apartado del **ANEXO C** respectivamente.

2.3.2.6. *Diseño de la Base de datos*

Para mantener la información ordenada, centralizada, accesible y actualizada se creará una base de datos que permita el acceso y manipulación de las diferentes funcionalidades que el sistema web requiera. Para identificar las entidades, atributos y relaciones de acuerdo con el funcionamiento del negocio se procedió a realizar un diagrama Entidad – Relación.

En base a este diagrama se obtiene el diseño del modelo lógico de la base de datos que cuenta con un total de 9 tablas, en las que se almacenará la información del negocio y permitirá un correcto almacenamiento y funcionamiento del sistema web.

Las tablas se encuentran identificadas por el nombre de la entidad o con la funcionalidad que va a cumplir dentro del sistema, además cada una de estas cuentan con sus respectivas claves primarias y claves foráneas, nótese que todas las claves primarias son de tipo int y auto incrementales y se encuentran presentes en todas las tablas.

Los demás tipos de datos que posee cada una de las tablas son los siguientes:

- **Int:** tipo de dato para especificar valores enteros.
- **Varchar:** tipo de dato para especificar, nombres, descripciones, etc.
- **Decimal:** tipo de dato para especificar valores monetarios.
- **Date:** tipo de dato para especificar fechas.

A continuación, en la **Figura 10-2**, se puede visualizar el diagrama lógico de la base de datos del sistema web, la misma que cuenta con 9 tablas cada una de ellas con sus respectivas claves primarias y foráneas con un total de 11 claves primarias y 7 claves foráneas, también cabe recalcar que la tabla mensaje posee atributos *de* y *para* que vienen de la *tabla administrador* y *cliente*, esta tabla no se encuentran relacionada porque en la *columna de* o *para* puede estar el *id* de la *tabla administrador* o *cliente*.

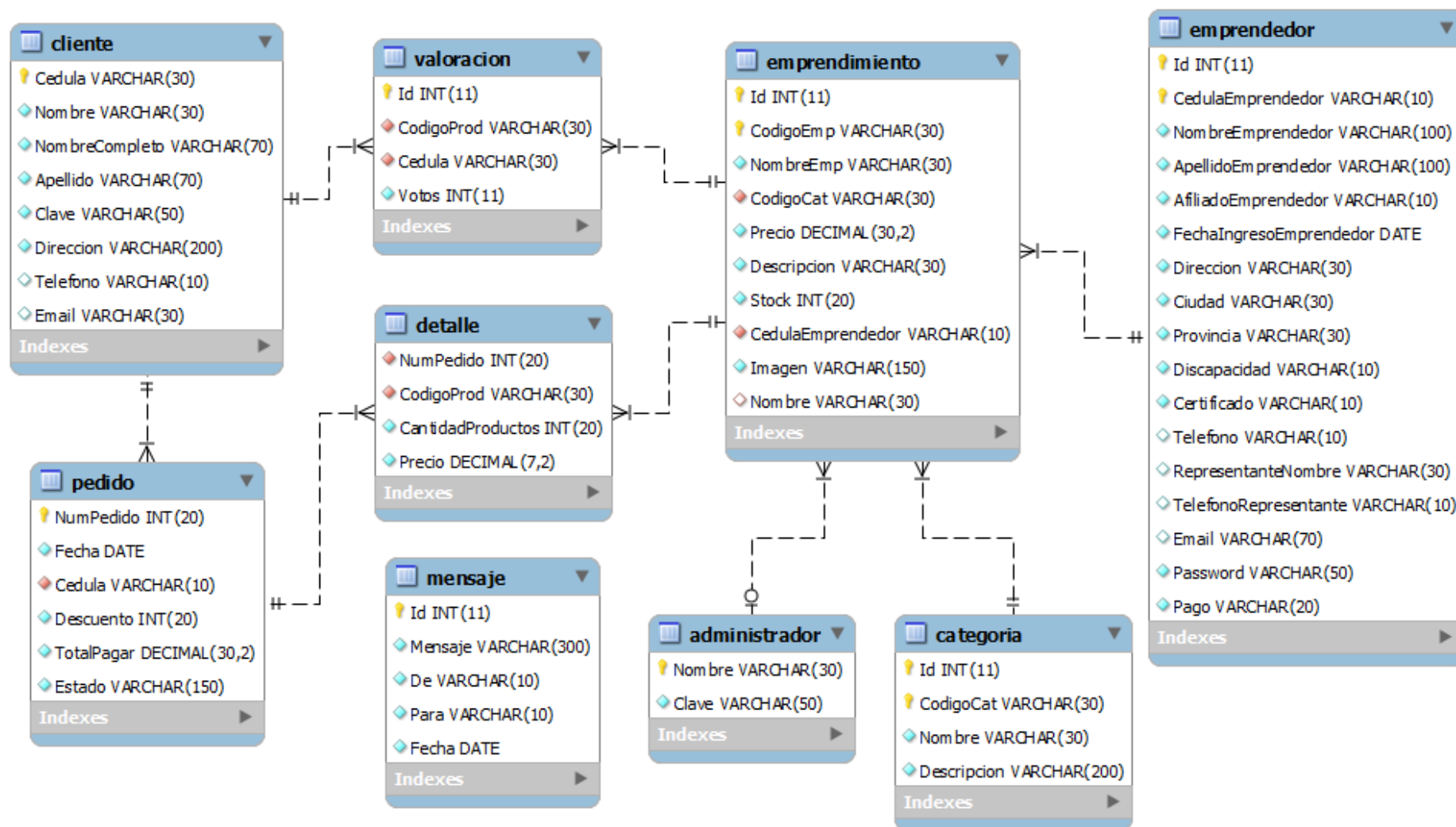


Figura 10-2: Modelo lógico de la base de datos.
 Realizado por: Adriana Quille, 2019.

2.3.2.7. Diccionario de datos

En el diccionario de datos de la BD presentada se especifican los diferentes tipos de datos que han sido utilizados para el manejo de la información que será subida a la base de datos respectivamente, todo esto con el objetivo de evitar todos aquellos posibles errores que puedan generarse en alguna actualización o mantenimiento que se realice al sistema, o también puede darse el caso de que pierda la estructura de la base de datos y poseer su respectivo respaldo del tipo de datos con el que se encontraba trabajando.

En la **Tabla 9-2**, puede observar el diccionario de datos de la tabla cliente con los diferentes tipos de datos que se han utilizado para el desarrollo del sistema, la misma que cuenta con una clave primaria y 8 campos de tipo varchar.

Tabla 9-2: Tabla cliente

	Clave primaria	Clave foránea	Tipo de dato	Tamaño	Nulo
Cedula	Si	No	varchar	30	No
Nombre	No	No	varchar	30	No
NombreCompleto	No	No	varchar	70	No
Apellido	No	No	varchar	70	No
Clave	No	No	varchar	50	No
Direccion	No	No	varchar	200	No
Telefono	No	No	varchar	10	Si
Email	No	No	varchar	70	Si

Realizado por: Adriana Quille, 2019.

También se definen las relaciones que existen con cada uno de los objetos para así poder mantener una idea clara y concisa con respecto al correcto funcionamiento de la información que posee el sistema. El diccionario de datos de la Fundación “Emprender y No Dependier” se encuentra en el **ANEXO D** donde podrá visualizar y analizar cada una de las dudas que se presenten.

2.3.2.8. Instalación, configuración y codificación

Para la codificación se utilizó el editor Visual Studio Code, Composer, MySQL y PHP. Es importante mencionar que para el desarrollo de la aplicación se utilizó el framework Symfony y el servidor apache para su correcto funcionamiento y para la parte del Chat en línea se desarrolló sobre la librería PHPWebSocket.

Para la realización de los respectivos reportes se utiliza DOMPDF siendo este una librería de PHP para la creación de documentos pdf.

Instalación de Composer

Para proceder con la instalación de composer previamente se debe tener instalado php (XAMPP).

1. Instalar el programa composer desde el link oficial de la página <https://getcomposer.org/download/>
2. Descargar el programa para Windows, como se muestra en la **Figura 11-2**.

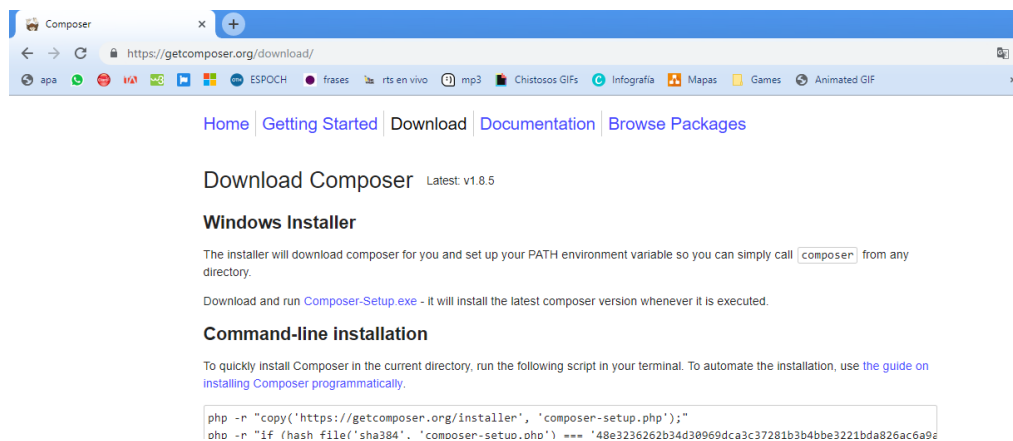


Figura 11-2: Descargar Composer para Windows.
Realizado por: Adriana Quille, 2019.

3. Una vez descargador el instalador, ejecutamos el archivo y damos clic en aceptar y presentará la siguiente pantalla **Figura 12-2**.

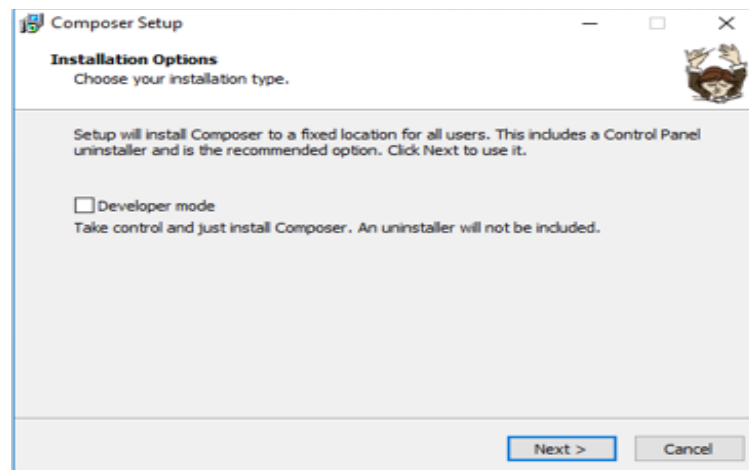


Figura 12-2: Instalación de Composer.
Realizado por: Adriana Quille, 2019.

4. Seleccionamos la dirección donde se encuentra instalado PHP, y damos clic en Next, procediendo así a la instalación como indica la **Figura 13-2** y damos clic en Next.

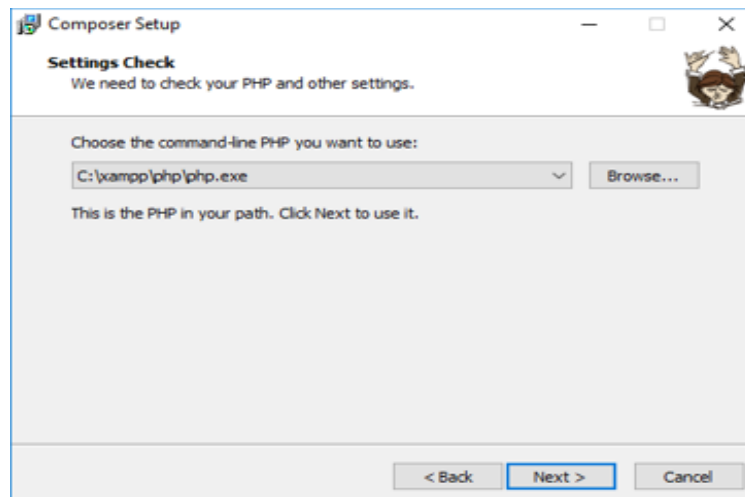


Figura 13-2: Seleccionamos la carpeta donde se encuentra instalado el PHP.
Realizado por: Adriana Quille, 2019.

5. Se presentará una ventana de instalación del composer como indica la **Figura 14-2** y damos click en Install.

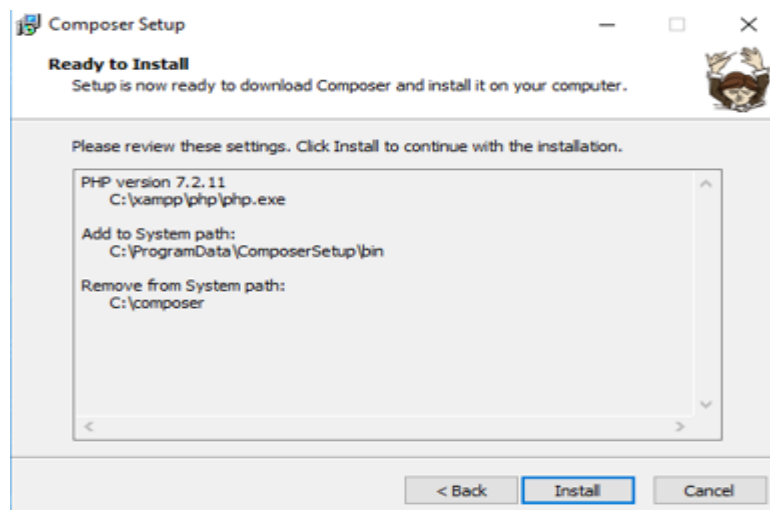


Figura 14-2: Instalación de Composer
Realizado por: Adriana Quille, 2019.

Instalación del framework Symfony

1. Para la instalación del framework Symfony deberá dirigirse a la documentación oficial en el sitio web <https://symfony.com/> en la cual indica que se debe instalar mediante composer la misma que ya se instaló anteriormente utilizando el comando:

composer create-project Symfony/website-skeleton + el nombre del Proyecto

Para esto se deberá abrir una terminal y ejecutarla ingresando la línea de comandos como indica la **Figura 15-2** y damos Enter.

```
C:\Windows\System32\cmd.exe
C:\xampp\htdocs> composer create-project symfony/website-skeleton fundacion_
```

Figura 15-2: Ingresando la línea de comandos.

Realizado por: Adriana Quille, 2019.

2. Posteriormente empezará a descargarse todos los componentes que Symfony ofrece como se puede visualizar en la **Figura 16-2**.

```
C:\Windows\System32\cmd.exe - composer create-project symfony/website-skeleton fundacion
C:\xampp\htdocs> composer create-project symfony/website-skeleton fundacion
Installing symfony/website-skeleton (v4.3.0.2)
- Installing symfony/website-skeleton (v4.3.0.2): Downloading (100%)
Created project in fundacion
Loading composer repositories with package information
Installing dependencies (including require-dev) from lock file
Package operations: 106 installs, 0 updates, 0 removals
- Installing ocradius/package-versions (1.4.0): Downloading (100%)
- Installing symfony/flex (v1.2.5): Downloading (100%)
```

Figura 16-2: Descarga todos los componentes que Symfony trae por defecto.

Realizado por: Adriana Quille, 2019.

3. En la **Figura 17-2**, se puede observar cómo se va instalando cada uno de los componentes de Symfony.

```
C:\Windows\System32\cmd.exe - composer create-project symfony/website-skeleton fundacion
- Installing symfony/flex (v1.2.5): Downloading (100%)
Prefetching 93 packages
- Downloading (100%)
- Installing doctrine/lexer (v1.0.1): Loading from cache
- Installing doctrine/annotations (v1.6.1): Loading from cache
- Installing doctrine/reflection (v1.0.0): Loading from cache
- Installing doctrine/event-manager (v1.0.8): Loading from cache
- Installing doctrine/collections (v1.6.1): Loading from cache
- Installing doctrine/cache (v1.8.0): Loading from cache
- Installing doctrine/persistence (1.1.1): Loading from cache
- Installing doctrine/inflector (v1.3.0): Loading from cache
- Installing doctrine/common (v2.10.0): Loading from cache
- Installing symfony/service-contracts (v1.1.2): Loading from cache
- Installing symfony/polyfill-abstract (v1.11.0): Loading from cache
- Installing symfony/doctrine-bridge (v4.3.0): Loading from cache
- Installing doctrine/doctrine-cache-bundle (1.3.5): Loading from cache
- Installing doctrine/instantiator (1.2.0): Loading from cache
- Installing symfony/stopwatch (v4.3.0): Loading from cache
- Installing symfony/polyfill-php73 (v1.11.0): Loading from cache
- Installing symfony/console (v4.3.0): Loading from cache
- Installing zendframework/zend-eventmanager (3.2.1): Loading from cache
- Installing zendframework/zend-code (3.3.1): Loading from cache
- Installing ocradius/proxy-manager (2.1.1): Loading from cache
- Installing doctrine/dbal (v2.9.2): Loading from cache
- Installing doctrine/migrations (v2.0.2): Loading from cache
- Installing egullas/email-validator (2.1.8): Loading from cache
- Installing jdorn/sql-formatter (v1.2.17): Loading from cache
```

Figura 17-2: Instalación de todos los componentes descargados anteriormente.

Realizado por: Adriana Quille, 2019.

4. Una vez finalizada la instalación de todos los componentes de Symfony se visualizará una pantalla como la **Figura 18-2**.

```
C:\Windows\System32\cmd.exe

* Run your application:
1. Go to the project directory
2. Create your code repository with the git init command
3. Download the Symfony CLI at https://symfony.com/download to install a development web server,
   or run composer require server --dev for a minimalist one

* Read the documentation at https://symfony.com/doc

Database Configuration

* Modify your DATABASE_URL config in .env
* Configure the driver (mysql) and
  server_version (5.7) in config/packages/doctrine.yaml

How to test?

* Write test cases in the tests/ folder
* Run php bin/phpunit

C:\xampp\htdocs>
C:\xampp\htdocs>
```

Figura 18-2: Instalación del Framework en el directorio fundación.
Realizado por: Adriana Quille, 2019.

5. Ahora se procede a ingresar al directorio donde se instaló el framework e ingresamos el comando: `php -S 127.0.0.1:8000 -t public` para empezar con nuestro servidor de desarrollo como indica la **Figura 19-2**.

```
C:\Windows\System32\cmd.exe - php -S 127.0.0.1:8000 -t public

C:\xampp\htdocs\fundacion>php -S 127.0.0.1:8000 -t public
PHP 7.2.11 Development Server started at Tue Jun  4 21:57:24 2019
Listening on http://127.0.0.1:8000
Document root is C:\xampp\htdocs\fundacion\public
Press Ctrl-C to quit.
```

Figura 19-2: Directorio del proyecto a desarrollar
Realizado por: Adriana Quille, 2019.

6. Ingresamos al navegador e ingresamos tipiendo `localhost:8000/` y deberá mostrarse la página de bienvenida del Framework Symfony acompañado de la versión en la que se trabajará, como se muestra en la **Figura 20-2**, si todo salió bien quiere decir que el framework se instaló correctamente, y está listo para empezar a trabajar en el desarrollo del sistema.

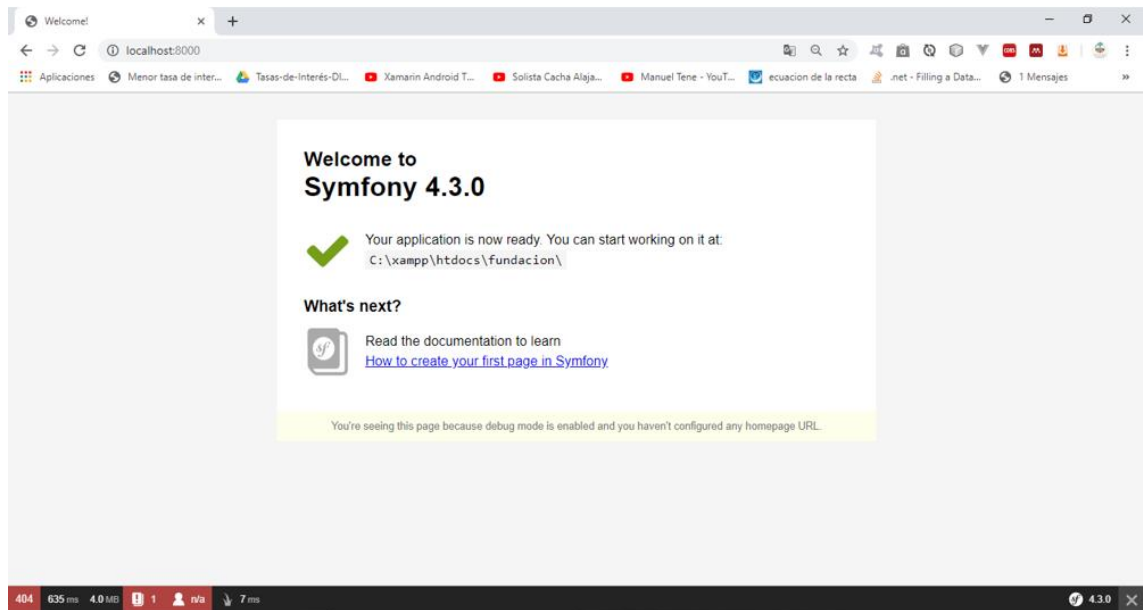


Figura 20-2: Página de bienvenida del framework Symfony.
Realizado por: Adriana Quille, 2019.

7. Accedemos al directorio `C:\xampp\htdocs\fundacion` para conocer la estructura del sistema como se muestra en la **Figura 21-2**, y cada uno de los directorios que va a reposar para cumplir con los requerimientos antes expuesto.

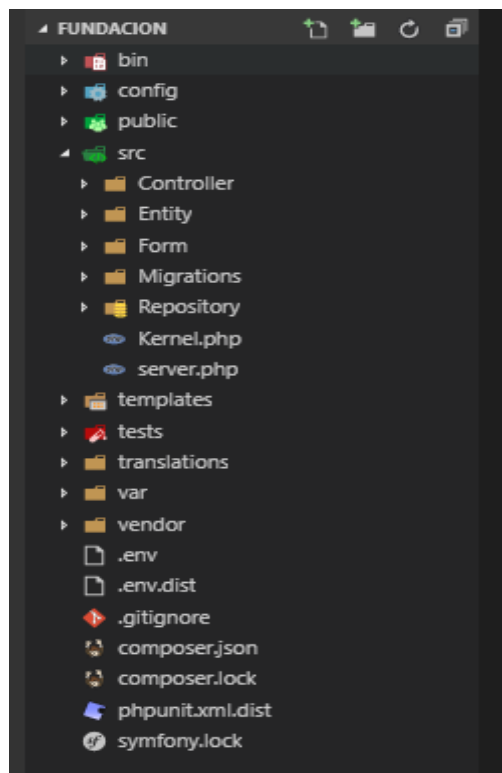


Figura 21-2: Estructura de las carpetas del sistema.
Realizado por: Adriana Quille, 2019.

A continuación, en la **Figura 22-2**, se puede observar la estructura del modelo (Entity) y controlador (Controller), del sistema

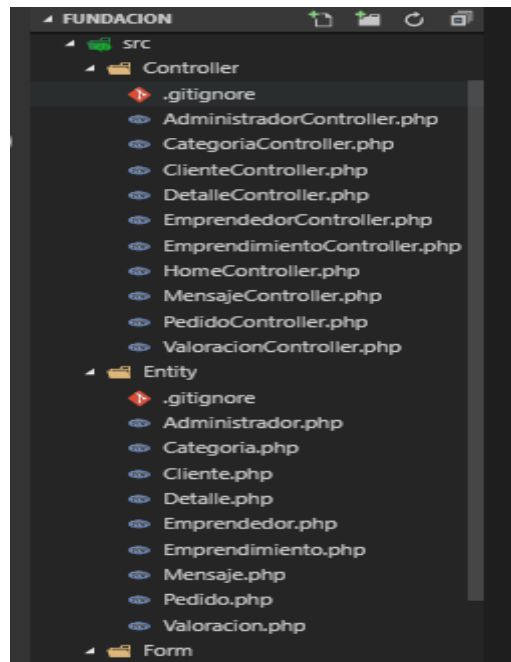


Figura 22-2: Estructura MVC del sistema
Realizado por: Adriana Quille, 2019.

En la carpeta Controller **Figura 23-2**, reposan los controller de todos los módulos a desarrollar cada uno de ellos con las respectivas acciones y restricciones.

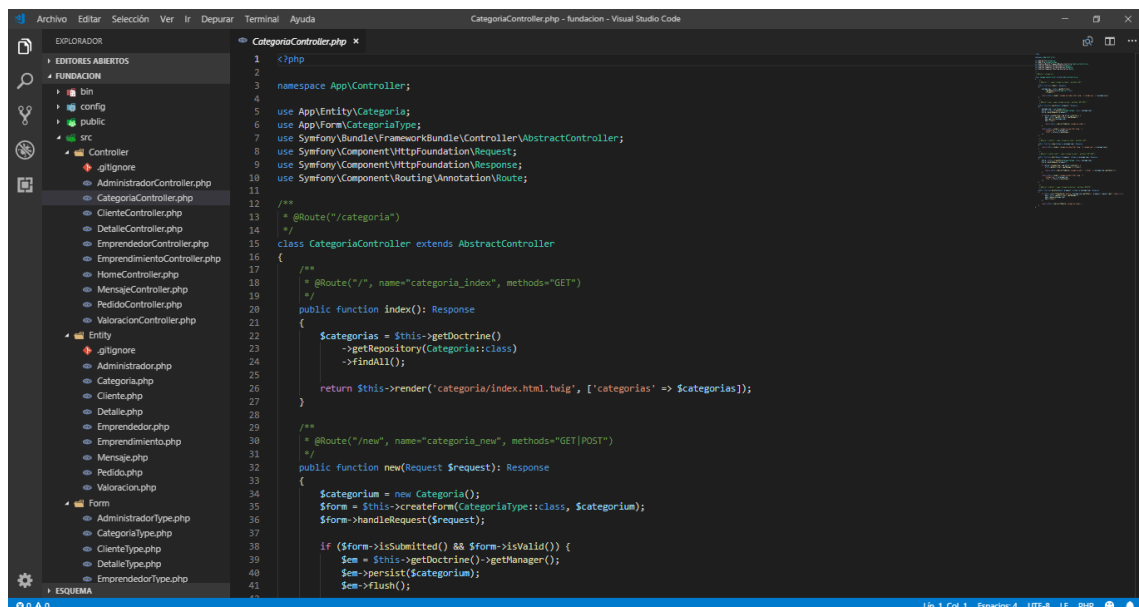
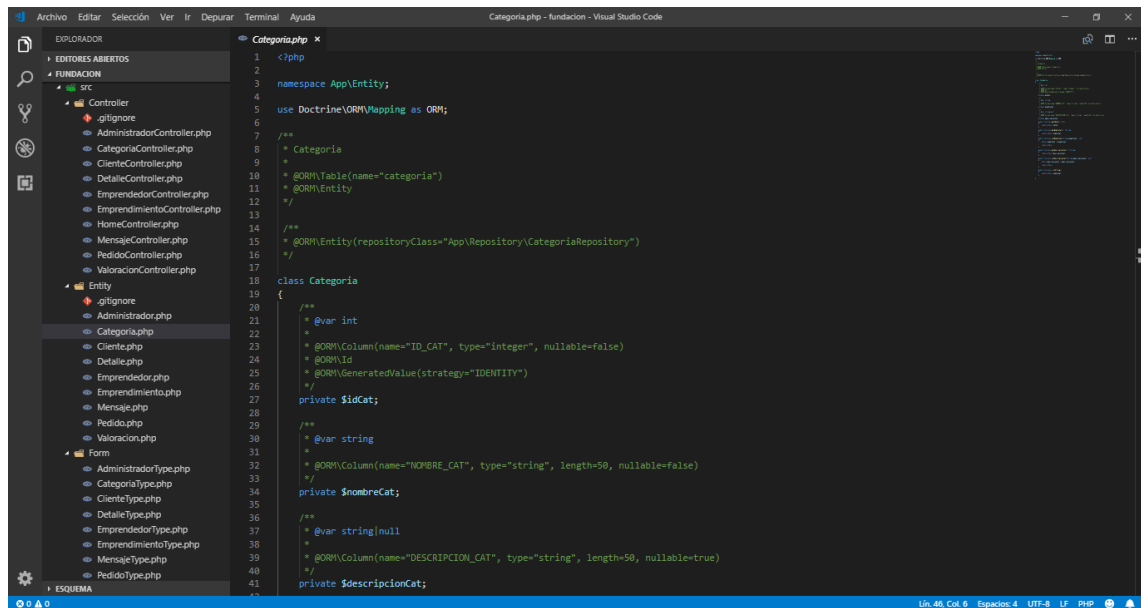


Figura 23-2: Controller de categorías del sistema.
Realizado por: Adriana Quille, 2019.

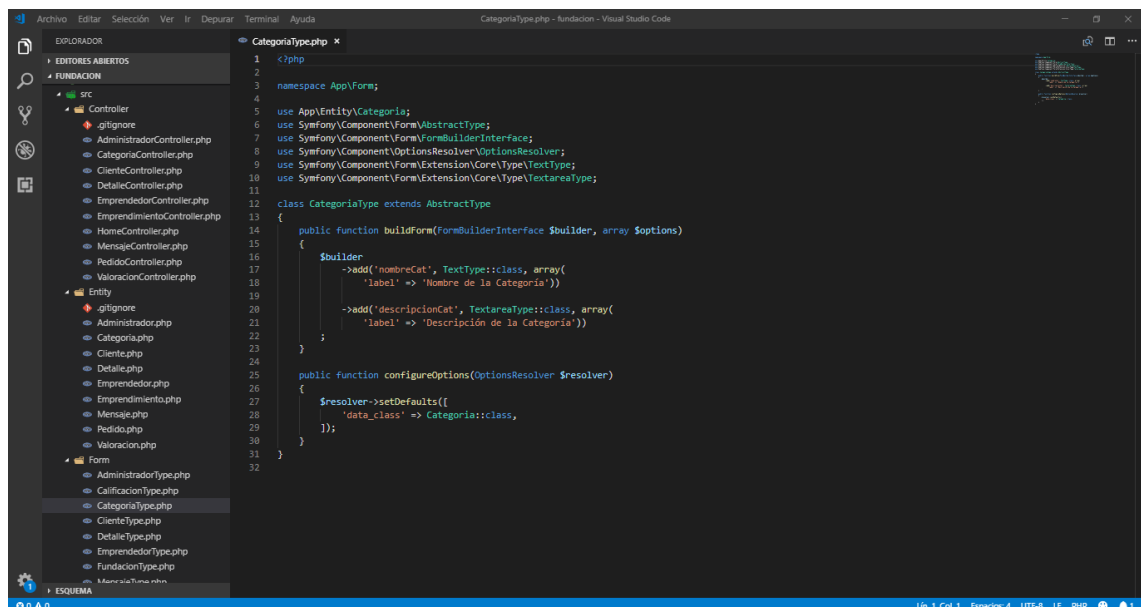
En la **Figura 24-2**, se puede observar la carpeta Entity, la misma que contiene los métodos get y set.



```
1 <?php
2 namespace App\Entity;
3 use Doctrine\ORM\Mapping as ORM;
4
5 /**
6  * Categoria
7  */
8 class Categoria
9 {
10     /**
11      * @ORM\Table(name="categoria")
12      * @ORM\Entity
13      */
14     /**
15      * @ORM\Entity(repositoryClass="App\Repository\CategoriaRepository")
16      */
17     class Categoria
18     {
19         /**
20          * @var int
21          */
22         /**
23          * @ORM\Column(name="ID_CAT", type="integer", nullable=false)
24          * @ORM\Id
25          * @ORM\GeneratedValue(strategy="IDENTITY")
26          */
27         private $idCat;
28
29         /**
30          * @var string
31          */
32         /**
33          * @ORM\Column(name="NOMBRE_CAT", type="string", length=50, nullable=false)
34          */
35         private $nombreCat;
36
37         /**
38          * @var string|null
39          */
40         /**
41          * @ORM\Column(name="DESCRIPCION_CAT", type="string", length=50, nullable=true)
42          */
43         private $descripcionCat;
```

Figura 24-2: Modelo de categoría del sistema
Realizado por: Adriana Quille, 2019.

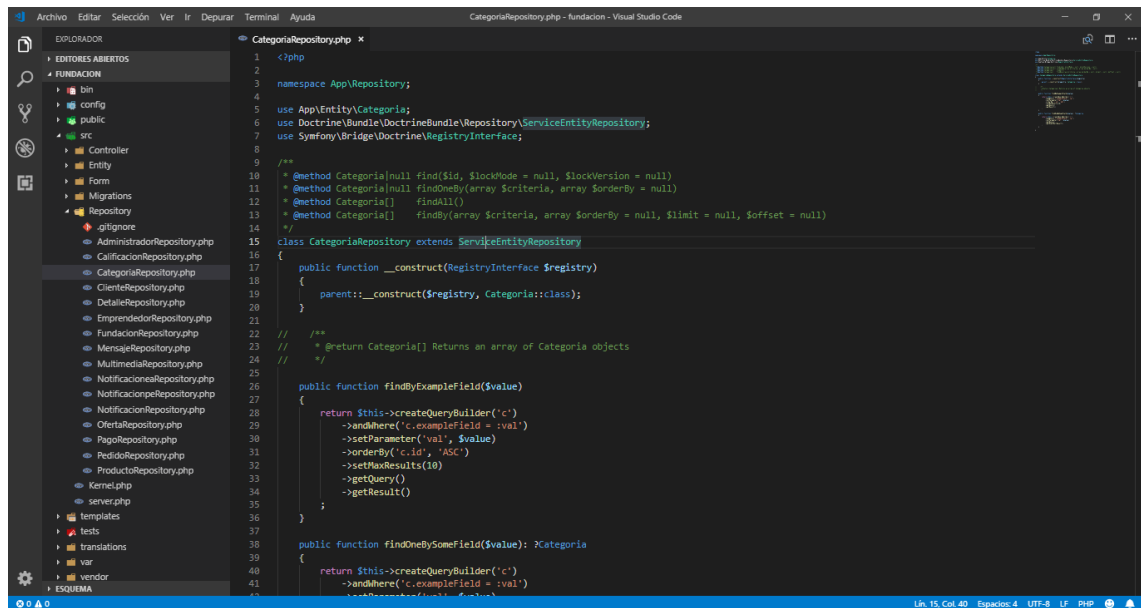
En la Carpeta Form reposan los formularios de las entidades, como se puede observar en la **Figura 25-2**.



```
1 <?php
2 namespace App\Form;
3
4 use App\Entity\Categoria;
5 use Symfony\Component\Form\AbstractType;
6 use Symfony\Component\Form\FormBuilderInterface;
7 use Symfony\Component\Form\FormBuilderInterface;
8 use Symfony\Component\OptionsResolver\OptionsResolver;
9 use Symfony\Component\Form\Extension\Core\Type\TextType;
10 use Symfony\Component\Form\Extension\Core\Type\TextareaType;
11
12 class CategoriaType extends AbstractType
13 {
14     public function buildForm(FormBuilderInterface $builder, array $options)
15     {
16         $builder
17             ->add('nombreCat', TextType::class, array(
18                 'label' => 'Nombre de la Categoría'))
19             ->add('descripcionCat', TextareaType::class, array(
20                 'label' => 'Descripción de la Categoría'));
21     }
22
23     public function configureOptions(OptionsResolver $resolver)
24     {
25         $resolver->setDefaults([
26             'data_class' => Categoria::class,
27         ]);
28     }
29 }
30
31
32
```

Figura 25-2: Formulario de la categoría del sistema.
Realizado por: Adriana Quille, 2019.

En la Carpeta Repository reposan las diversas consultas para acceder a la base de datos como se puede observar en la **Figura 26-2**.



```
1 <?php
2 namespace App\Repository;
3
4 use App\Entity\Categoria;
5 use Doctrine\Bundle\DoctrineBundle\Repository\ServiceEntityRepository;
6 use Symfony\Bridge\Doctrine\RegistryInterface;
7
8 /**
9  * @method Categoria|null find($id, $lockMode = null, $lockVersion = null)
10  * @method Categoria|null findOneBy(array $criteria, array $orderBy = null)
11  * @method Categoria[] findAll()
12  * @method Categoria[] findBy(array $criteria, array $orderBy = null, $limit = null, $offset = null)
13
14 */
15 class CategoriaRepository extends ServiceEntityRepository
16 {
17     public function __construct(RegistryInterface $registry)
18     {
19         parent::__construct($registry, Categoria::class);
20     }
21
22     /**
23      * @return Categoria[] Returns an array of Categoria objects
24      */
25
26     public function findByExampleField($value)
27     {
28         return $this->createQueryBuilder('c')
29             ->andWhere('c.exampleField = :val')
30             ->setParameter('val', $value)
31             ->orderBy('c.id', 'ASC')
32             ->setMaxResults(10)
33             ->getQuery()
34             ->getResult();
35     }
36
37     public function findOneBySomeField($value): ?Categoria
38     {
39         return $this->createQueryBuilder('c')
40             ->andWhere('c.exampleField = :val')
41             ->setParameter('val', $value)
42             ->getQuery()
43             ->getResult();
44     }
45 }
```

Figura 26-2: Repository del sistema.
Realizado por: Adriana Quille, 2019.

Cabe mencionar que para cada uno de los módulos se encuentra desarrollado de la misma manera como lo visualizamos del módulo de categorías, para cumplir con el patrón de diseño MVC utilizando el framework Symfony.

Instalación de la tecnología PHPWebSocket

1. Las funciones de socket son parte de una extensión de PHP que se debe habilitar para el momento de compilación con la línea de comando **--enable-sockets** para la respectiva configuración.
2. Luego accedemos al archivo de php.ini que lo encontramos en el directorio de XAMPP **C:\xampp\php** y procedemos a des comentar la línea de código como puede observar en la **Figura 27-2**, borrando el punto y coma (;) y clic en guardar los cambios.

```

php: Bloc de notas
Archivo Edición Formato Ver Ayuda
;extension=php_pdo_oci.dll
;extension=php_pdo_odbc.dll
;extension=php_pdo_pgsql.dll
extension=php_pdo_sqlite.dll
;extension=php_pgsql.dll
;extension=php_shmop.dll

; The MIBS data available in the PHP distribution must be installed.
; See http://www.php.net/manual/en/snmp.installation.php
;extension=php_snmp.dll

;extension=php_soap.dll
extension=php_sockets.dll
;extension=php_sqlite3.dll

```

Figura 27-2: Habilitando la extensión socket.
Realizado por: Adriana Quille, 2019.

- Posteriormente creamos las funciones para que el socket pueda funcionar correctamente quienes serán los encargados de devolver recursos socket, los mismos que se encuentran en el archivo *server.php* como se puede observar en la **Figura 28-2**.

```

server.php - fundacion - Visual Studio Code
server.php x edit.html.twig
1 <?php
2 function wsOnMessage($clientID, $message, $messageLength, $binary) {
3     global $Server;
4     $objMensaje = json_decode($message);
5     $texto=$objMensaje->('Message');
6     $de=$objMensaje->('De');
7     $para=$objMensaje->('Para');
8     include "process/chat/InsertarMensajes.php";
9     $ip = long2ip( $Server->wsClients[$clientID][6] );
10    if ( $messageLength == 0 ) {
11        $Server->wsClose($clientID);
12    }
13    }
14
15    if ( sizeof($Server->wsClients) == 1 )
16        $Server->wsSend($clientID, $listaMensajesRetornar);
17    else
18        foreach ( $Server->wsClients as $id => $client )
19            $Server->wsSend($id, $listaMensajesRetornar);
20
21    function wsOnOpen($clientID){
22        global $Server;
23        $ip = long2ip( $Server->wsClients[$clientID][6] );
24        $Server->log( " $ip ( $clientID ) has connected." );
25        foreach ( $Server->wsClients as $id => $client )
26            if ( $id != $clientID )
27                $Server->wsSend($id, "Visitor $clientID ( $ip ) has joined the room.");
28    }
29
30    function wsOnClose($clientID, $status) {
31        global $Server;
32        $ip = long2ip( $Server->wsClients[$clientID][6] );
33        $Server->log( " $ip ( $clientID ) has disconnected." );
34        foreach ( $Server->wsClients as $id => $client )
35            $Server->wsSend($id, "Visitor $clientID ( $ip ) has left the room.");
36    }
37
38    $Server = new PHPWebSocket();
39    $Server->bind("message", "wsOnMessage");
40    $Server->bind("open", "wsOnOpen");
41    $Server->bind("close", "wsOnClose");
42    $Server->wsStartServer("127.0.0.1", 90);

```

Figura 28-2: Creación el servidor socket.
Realizado por: Adriana Quille, 2019.

2.3.2.9. Historias de usuario

Las historias de usuarios son aquellas que representan los requerimientos que han sido planteados para el desarrollo del sistema web por parte del cliente, todos estos requerimientos deben estar respectivamente documentadas para dar a conocer todo lo que ha realizado con sus respectivos criterios y pruebas de aceptación en cada historia de usuario, también cuentan con tareas de ingenierías que también deberán estar documentadas.

Las historias de usuario están compuestas por:

- **ID o Número:** identificador único para cada tarea o actividad, el formato que se seguirá es **HU** (Historia de Usuario) o **HT** (Historia Técnica) continuando con la respectiva numeración que deberá constar de dos dígitos.
- **Nombre:** nombre de la tarea o actividad que se va a realizar.
- **Modificación de la HU:** permite conocer si la **HU** o **HT** ha sido modificada y en caso de serlo se coloca la nueva numeración modificada.
- **Usuario:** usuario encargado de acceder a la funcionalidad para verificarlo.
- **Sprint Asignado:** número del Sprint a la pertenece la tarea o actividad.
- **Prioridad del negocio:** se asigna: Alta, Media o Baja, de acuerdo a las necesidades del cliente.
- **Riesgo del desarrollo:** se asigna: Alta, Media o Baja, de acuerdo al grupo de desarrollo.
- **Puntos estimados:** tiempo que se requiere para el desarrollo de la actividad o tarea respectivamente.
- **Puntos reales:** tiempo real que se tomó en realizar la tarea o actividad.
- **Descripción:** se describe lo que se requiere y el beneficio para el que va a ser desarrollado.
- **Observación:** describe datos importantes para el desarrollo de la tarea o actividad.

Reverso de la HU: se describe todas las pruebas de aceptación siendo estas las condiciones que deberá cumplir la tarea o actividad que ha sido desarrollada para su correcto funcionamiento, estas pruebas de aceptación deben estar documentadas.

A continuación, en las *tablas* que se detallan se presenta la documentación de una historia de usuario como ejemplo de la manera en la que se ha documentado las demás historias de usuario que forman parte del sistema.

Tabla 10-2: Historia de usuario 01: Registrar datos de los productos.

HISTORIA DE USUARIO	
Número: HU_01	Nombre: Registrar datos de los productos
Modificación historia de usuario:	
Usuario: Emprendedor	Sprint Asignada: 2
Prioridad del Negocio: Alta	Puntos Estimados: 20
Riesgo de desarrollo: Alta	Puntos Reales: 20
Descripción: Como emprendedor, necesito ingresar los datos de un producto con la finalidad de tener un registro con la información real y poder realizar cualquier gestión con el producto que se requiera.	
Observación:	
Pruebas de Aceptación:	

- Verificar si todos los registros ingresados en el formulario han sido guardados correctamente.

Realizado por: Adriana Quille, 2019.

Tabla 11-2: Prueba de aceptación de la HU_01.

PRUEBA DE ACEPTACIÓN	
Código: PA_01	Nombre de la Historia: Registrar datos de los emprendimientos
Nombre de la Prueba: Verificar si todos los registros ingresados en el formulario han sido guardados correctamente.	
Responsable: Adriana Quille	Fecha: 30/11/2018
Descripción: Se verificará si todos los campos ingresados han sido guardados en la base de datos.	
Condición de Ejecución:	
<ul style="list-style-type: none"> • La base de datos debe estar creada. • El nuevo emprendimiento no debe existir en la base de datos. 	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Ingresar al apartado de emprendimientos. 2. En el apartado de agregar emprendimiento llenamos todos los registros que requiera el formulario y damos clic en Añadir. 3. Visualizará un mensaje de datos guardados correctamente. 4. Dirigirse a la base de datos y constatar los datos registrados correctamente. 	
Resultado Esperado: Se debe presentar un mensaje de datos guardados correctamente.	
Evaluación de la Prueba: Exitosa	

Realizado por: Adriana Quille, 2019.

Tabla 12-2: Tarea de ingeniería 01 de la HU_01.

TAREA DE INGENIERÍA	
Sprint: 1	Número de Tarea: 01
Nombre de la Historia: Registrar datos de los emprendimientos	
Nombre de la Tarea: Acceso a datos para el registro de datos de los emprendimientos.	
Responsable: Adriana Quille	Tipo de Tarea: Desarrollo
Fecha Inicio: 28/11/2018	Fecha Fin: 28/11/2018
Descripción: Se creará el archivo para el acceso a datos y la respectiva sentencia SQL.	
Prueba de Aceptación:	
<ul style="list-style-type: none"> • Verificar que exista conexión con la base de datos. 	

Realizado por: Adriana Quille, 2019.

Para las demás historias de usuario que han sido desarrollados deberá dirigirse al **ANEXO E** donde se encuentra la respectiva documentación del sistema web.

2.3.2.10. Manual de usuario

El manual de usuario es uno de los elementos más importantes en la entrega de desarrollo de software a través de este se explica detalle a detalle el funcionamiento del software, en este manual se incluye desde el modo de acceso, ingreso, actualización, consultas, reportes o eliminación de información de acuerdo con lo que se requiere realizar. Este manual de usuario lo puede visualizar detalladamente en el **ANEXO F**.

2.3.3. Fase de Finalización

En esta fase de finalización se presenta las actividades que se realizó para la culminación de este proyecto técnico, en la **Tabla 13-2**, también para poner en funcionamiento para su utilización por el cliente.

Tabla 13-2: Actividades de finalización del proyecto.

Actividad	Descripción	Responsable
Documentación del sistema	<ul style="list-style-type: none"> Manual Técnico Manual de Usuario 	Adriana Quille
Capacitación de los Usuarios	Entrega del manual de usuario y técnico al cliente (Product Owner).	Adriana Quille

Realizado por: Adriana Quille, 2019.

2.3.3.1. Burndown Chart

Una vez finalizado la parte de desarrollo con los diferentes Sprint con los que cuenta el sistema web podemos visualizar el **Gráfico 1-2** donde reposa el Burndown chart, herramienta que ayuda a los miembros del equipo de desarrollo a ir observando la velocidad con la que ha avanzado el sistema que se encuentra en desarrollo y así conocer si se encuentra cumpliendo los objetivos planificados con sus respectivos tiempos establecidos.

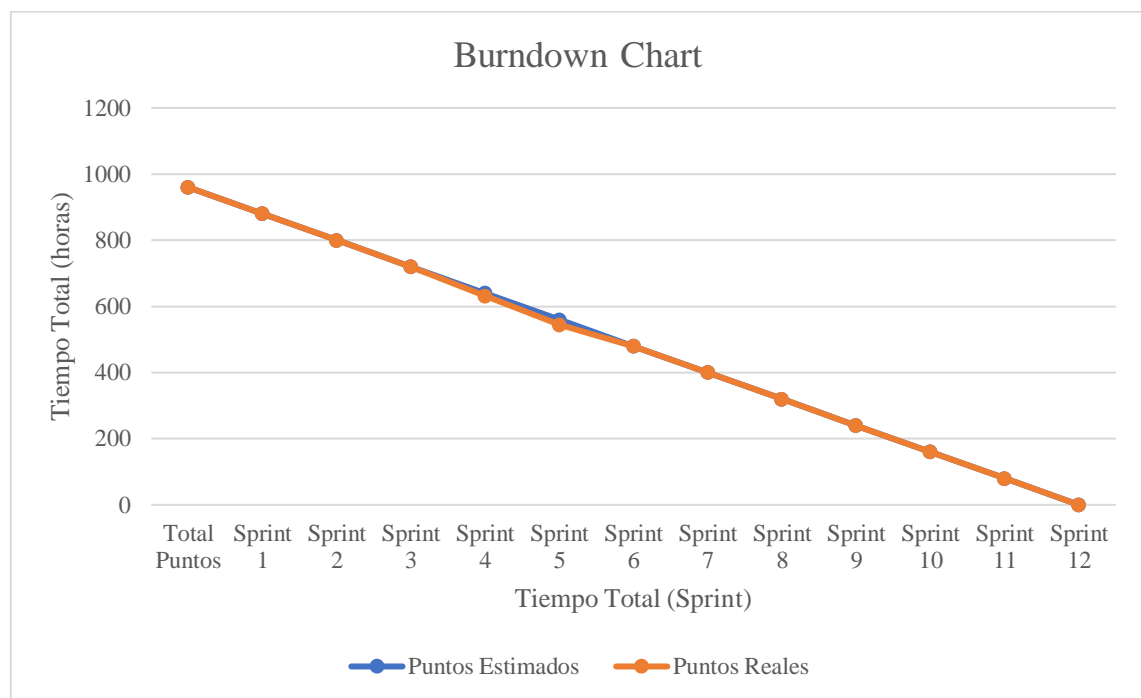


Gráfico 1-2: Burndown Chart

Realizado por: Adriana Quille, 2019.

En el **Gráfico 1-2**, se puede observar dos líneas, una de color azul que refleja el camino a seguir según lo estimado y planificado, mientras que la línea de color naranja indica el rumbo que toma el desarrollo del proyecto en realidad, nótese que presenta variaciones en ambas líneas dando a conocer que en el sprint 5 se ha tomado más tiempo de lo estimado; pero cabe recalcar que el desarrollo del sistema web no se vio afectado ante estos inconvenientes presentados, en el sprint 9 se sobreestimo a las tareas planificadas puesto que el tiempo del sprint 5 se recompensó con el sprint antes mencionado.

2.4. Métodos utilizados para evaluar la eficiencia del sistema web

2.4.1. *Análisis de la eficiencia del sistema*

Para evaluar la eficiencia del sistema se efectúa un estudio comparativo basado en los tiempos de respuesta del sistema en la ejecución de los procesos realizados por los clientes de la Fundación “Emprender y No Dependier” en el registro de pedidos y en los procesos de registro de los emprendimientos y registro del emprendedor de los trabajadores. Para su evaluación se utiliza una comparativa con los tiempos de respuesta de forma tradicional y sistematizada.

2.4.2. *Población*

Para medir el parámetro de calidad de eficiencia del sistema web se tomó una población de 41 requerimientos funcionales conocidos como historias de usuario, dado que el software está orientado específicamente para la Fundación “Emprender y No Dependier”; se trabaja con esta población porque no se conoce el número de usuarios por cada rol que manipularán dicho sistema.

2.4.3. *Muestra*

La muestra para evaluar la eficiencia es de 3 requerimientos funcionales que forman parte del sistema web; estos requerimientos funcionales han sido tomados aplicando un muestreo no probabilístico o por conveniencia, debido a que se toma los requerimientos más utilizados por cada rol (Administrador, emprendedor y cliente).

Para evaluar el parámetro de la eficiencia se tomó los respectivos tiempos durante una semana en la fundación “Emprender y No Dependier”; entre los procesos tradicionales y los procesos sistematizados, para ello se usó la técnica de la observación con la ayuda de un cronómetro.

A continuación, los procesos que van a ser evaluados son los siguientes:

- **Registrar emprendimientos**

El proceso del “registro de emprendimientos” consiste en agregar al listado de emprendimientos que la Fundación ofrece a los clientes, para esto el emprendedor debe cumplir con varios parámetros que la fundación tiene establecido. Estos emprendimientos se encuentran organizados por categorías.

- **Registrar emprendedores**

El proceso del “registro de emprendedores” consiste en agregar al listado de emprendedores que pertenecen a la Fundación con su respectivo emprendimiento, los mismos que deben ser calificados por la fundación en base a las reglas que la fundación establece.

- **Registrar pedido**

El proceso del “registro de pedidos” consiste en ingresar la fecha en la que se realiza dicha petición de un cliente y el número de pedido con el que es aprobado.

La métrica por analizar en el presente trabajo de titulación es la eficiencia; en el cual se evaluará el comportamiento de tiempos de respuesta de los procesos de registrar pedido, emprendimientos y emprendedor respectivamente donde, entre menor sea el tiempo de respuesta, mejor será para el cliente.

2.4.4. Comportamiento de tiempos

Se desarrollo un estudio comparativo, el primero dirigido a los clientes y el segundo a los trabajadores de la Fundación “Emprender y No Dependier”. La tabla con los tiempos obtenidos en segundos se puede visualizar en el **Anexo G**. Las tareas expuestas en el apartado **2.4.3**. son las que se han tomado en cuenta para el comportamiento de tiempos.

2.4.5. Obtención de datos

Para el proceso de obtención de datos se realiza a partir del registro de pedidos, del emprendimiento y emprendedor aplicando la técnica de la observación más el uso de un cronómetro como se indica en el apartado **2.1.3**. de las técnicas de investigación.

Cabe mencionar que para la recopilación de los tiempos de los procesos tradicionales la Presidenta de la fundación inicialmente llena los datos del pedido, emprendimiento y emprendedor, de forma manual y para la obtención de los tiempos de los procesos sistematizados se hace uso del sistema web realizando los mismos procesos que anteriormente se realizó manualmente.

Los tiempos de respuesta de los procesos expuestos en el apartado **2.4.3.**, se tomaron a partir del lunes 6/5/2019 hasta el sábado 11/5/2019 en horarios de 09:00 a 12:30 y de 14:00 a 18:00. Para obtener los tiempos de respuesta de los procesos a analizar de forma sistematizada se utilizó una laptop TOSHIBA, procesador Intel Core i7, con memoria RAM de 8.0 GB y 1 tera de disco duro.

Posterior a la toma de tiempos de respuesta se recopilaron un total de 30 tiempos para el registro de pedidos, 15 tiempos para el registro de emprendimientos, y 10 tiempos para el registro de emprendedores, estos tiempos fueron tomados en segundos. Los tiempos obtenidos se puede observar en el **Anexo G.**

Con la finalidad de conocer si los tiempos de respuesta que se obtuvieron en los procesos mencionados en el apartado **2.4.3.** provienen de una distribución normal se procede a realizar el análisis de normalidad de los datos para establecer posteriormente que distribución de probabilidad aplicar.

CAPÍTULO III

3. MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS

En este capítulo se muestra la evaluación realizada al sistema web para la Fundación “Emprender y No Dependier” haciendo uso del estándar 9126 con respecto al análisis de la eficiencia mediante el comportamiento de tiempos en el sistema desarrollado, estableciendo indicadores en base a la situación actual del mismo con el fin de realizar cada una de las actividades requeridas por el cliente para comprobar el cumplimiento del objetivo planteado en este trabajo de titulación.

3.1.1. *Análisis de la normalidad de los datos*

Para evaluar la normalidad de los tiempos obtenidos de los procesos expuestos en el apartado 2.4.3 se utilizó el Software estadístico RStudio.

1. Planteamiento de la hipótesis:

Hipótesis Nula (H_0) = Los datos provienen de una distribución normal.

Hipótesis Alterna (H_1) = Los datos no provienen de una distribución normal.

2. Nivel de Significancia

El nivel de significancia es de $\alpha = 0.05$ considerado como un margen de error mínimo garantizando el 95% de confiabilidad.

3. Estadístico de Prueba

Se usó Shapiro-Wilk test para comprobar si los tiempos de respuesta tradicionales y sistematizados que han sido recopilados provienen de una distribución normal, dado el caso que la muestra es pequeña.

A continuación, en las *figuras* se muestran los tiempos tradicionales y sistematizados de cada proceso a evaluar.

- Registro de pedidos

En la **tabla 1-3** se puede observar los tiempos tradicionales del registro de pedidos y en la **tabla 2-3** los tiempos sistematizados del mismo.

```

212
213 RegistroPedidosTradicional<-c(196, 195, 194, 193, 197, 193,
214                               197, 198, 204, 191, 198, 204,
215                               195, 197, 200, 193, 196, 196,
216                               191, 197, 196, 190, 193, 195,
217                               201, 199, 204, 195, 194, 197)
218 shapiro.test(RegistroPedidosTradicional)
219
220
221 <
221:1 (Top Level) ↕

```

```

Console ~/
>
>
> RegistroPedidosTradicional<-c(196, 195, 194, 193, 197, 193,
+                               197, 198, 204, 191, 198, 204,
+                               195, 197, 200, 193, 196, 196,
+                               191, 197, 196, 190, 193, 195,
+                               201, 199, 204, 195, 194, 197)
> shapiro.test(RegistroPedidosTradicional)

      Shapiro-wilk normality test

data:  RegistroPedidosTradicional
W = 0.93968, p-value = 0.08914

```

Figura 1-3: Tiempos tradicionales del registro de pedidos
Realizado por: Adriana Quille, 2019.

```

229 RegistroPedidosSistematizado<-c(66, 65, 66, 67, 65, 64,
230                                 66, 64, 63, 66, 63, 64,
231                                 67, 68, 65, 63, 66, 67,
232                                 65, 67, 65, 69, 65, 63,
233                                 63, 64, 63, 65, 64, 66)
234 shapiro.test(RegistroPedidosSistematizado)
235
236 <
238:1 (Top Level) ↕

```

```

Console ~/
>
>
> RegistroPedidosSistematizado<-c(66, 65, 66, 67, 65, 64,
+                                 66, 64, 63, 66, 63, 64,
+                                 67, 68, 65, 63, 66, 67,
+                                 65, 67, 65, 69, 65, 63,
+                                 63, 64, 63, 65, 64, 66)
> shapiro.test(RegistroPedidosSistematizado)

      Shapiro-wilk normality test

data:  RegistroPedidosSistematizado
W = 0.93303, p-value = 0.05913

```

Figura 2-3: Tiempos sistematizados del registro de pedidos
Realizado por: Adriana Quille, 2019.

- Registrar de datos de los emprendimientos

En la **tabla 3-3** se puede observar los tiempos tradicionales del registro de datos de los emprendimientos y en la **tabla 4-3** los tiempos sistematizados del mismo.

```
242
243 RegistroEmprendimientosTradicional<-c(218, 238, 209, 237, 215,
244                                     197, 199, 239, 191, 221,
245                                     196, 215, 201, 229, 236)
246 shapiro.test(RegistroEmprendimientosTradicional)
247 |
248
249 <
247:1 (Top Level) ↕
```

```
Console ~/ ↵
>
>
>
>
> RegistroEmprendimientosTradicional<-c(218, 238, 209, 237, 215,
+                                     197, 199, 239, 191, 221,
+                                     196, 215, 201, 229, 236)
> shapiro.test(RegistroEmprendimientosTradicional)

      shapiro-wilk normality test

data:  RegistroEmprendimientosTradicional
w = 0.91468, p-value = 0.1597
```

Figura 3-3: Tiempos tradicionales del registro de datos de los emprendimientos
Realizado por: Adriana Quille, 2019.

```
255 RegistroEmprendimientosSistematizado<-c(123, 123, 121, 123, 122,
256                                     120, 121, 120, 124, 124,
257                                     125, 124, 121, 122, 124)
258 shapiro.test(RegistroEmprendimientosSistematizado)
259 |
260
261 <
259:1 (Top Level) ↕
```

```
Console ~/ ↵
>
>
> RegistroEmprendimientosSistematizado<-c(123, 123, 121, 123, 122,
+                                     120, 121, 120, 124, 124,
+                                     125, 124, 121, 122, 124)
> shapiro.test(RegistroEmprendimientosSistematizado)

      shapiro-wilk normality test

data:  RegistroEmprendimientosSistematizado
w = 0.92436, p-value = 0.2244
```

Figura 4-3: Tiempos sistematizados del registro de datos de los emprendimientos
Realizado por: Adriana Quille, 2019.

- **Registro de datos de los emprendedores**

En la **tabla 5-3** se puede observar los tiempos tradicionales del registro de datos de los emprendedores y en la **tabla 6-3** los tiempos sistematizados del mismo.

```
264 RegistroEmprendedorTradicional<-c(165, 128, 121, 120, 164,
265                                     128, 179, 163, 159, 177)
266 shapiro.test(RegistroEmprendedorTradicional)
267 |
268 <
269 <
267:1 (Top Level) ↕
```

```
Console ~/ ↵
>
> RegistroEmprendedorTradicional<-c(165, 128, 121, 120, 164,
+                                     128, 179, 163, 159, 177)
> shapiro.test(RegistroEmprendedorTradicional)

      shapiro-wilk normality test

data:  RegistroEmprendedorTradicional
W = 0.84813, p-value = 0.05518
```

Figura 5-3: Tiempos tradicionales del registro de emprendedores
Realizado por: Adriana Quille, 2019.

```
270 RegistroEmprendedorSistematizado<-c(82, 95, 91, 69, 76,
271                                       104, 96, 62, 89, 79)
272 shapiro.test(RegistroEmprendedorSistematizado)
273 |
274 <
275 <
273:1 (Top Level) ↕
```

```
Console ~/ ↵
>
> RegistroEmprendedorSistematizado<-c(82, 95, 91, 69, 76,
+                                       104, 96, 62, 89, 79)
> shapiro.test(RegistroEmprendedorSistematizado)

      shapiro-wilk normality test

data:  RegistroEmprendedorSistematizado
W = 0.97774, p-value = 0.952
```

Figura 6-3: Tiempos sistematizados del registro de emprendedores
Realizado por: Adriana Quille, 2019.

Los tiempos se encuentran expresados en segundos y almacenados en un vector tanto para el proceso tradicional como para el sistematizado y para su análisis se utilizó el software RStudio, luego se aplicó Shapiro Wilk test normality obteniendo el resultado de p-value mayor a 0.05, estos resultados se utilizarán para establecer la regla de decisión.

4. Establecer regla de decisión

Haciendo uso del valor de p (*p-value*) obtenido en el punto 3, se estableció la regla de decisión $p > \alpha$, donde se concluye que la condición establecida cumple por tanto *No se rechaza H_0* , dando como resultado que los tiempos de los procesos expresados en segundos provienen de una

distribución normal. A continuación, en la **Tabla 1-3**, se detalla cada uno de los tiempos expuestos anteriormente en punto 3.

Tabla 1-3: Resultados de tiempos en segundos para la toma de decisión

	Tiempo Tradicional (segundos)		Tiempo Sistematizado (segundos)	
	$p > \alpha$		$p > \alpha$	
Registro de pedidos	0,08914 > 0,05	No rechazo H_0	0,05913 > 0,05	No rechazo H_0
Registro de datos de los emprendimientos	0,1597 > 0,05	No rechazo H_0	0,2244 > 0,05	No rechazo H_0
Registro de datos de los emprendedores	0,05518 > 0,05	No rechazo H_0	0,952 > 0,05	No rechazo H_0

Realizado por: Adriana Quille, 2019.

5. Toma de decisión

Dado el valor de p (p -value) de los procesos evaluados, se puede evidenciar que es mayor al nivel de significancia de 0.05 por lo cual no se rechaza H_0 , concluyendo que los datos analizados vienen de una distribución normal.

3.1.2. Caso 1

El caso uno se encuentra dirigido a los clientes de la Fundación “Emprender y No Dependier”, se analizó la funcionalidad más significativa del sistema para el rol cliente, siendo este el registro de pedidos.

3.1.2.1. Estadística Descriptiva

Se aplicó la estadística descriptiva para el análisis de forma general de los tiempos que fueron recopilados y posteriormente realizar la respectiva comparación, cabe recalcar que los tiempos analizados se encuentran en el **Anexo G**.

- **Registro de pedidos**

Para el proceso de registrar pedidos se obtuvo un total de 30 tiempos expresados en segundos, los mismos que permitieron evaluar la eficiencia del sistema, se tomó en cuenta la *media*, *desviación estándar*, *valor mínimo* y *máximo* de los tiempos obtenidos durante este proceso, en los cuales se obtuvo un tiempo promedio de 196,3 segundos en el método tradicional, mientras que con el método sistematizado se obtuvo un tiempo promedio de 65,13 segundos, la desviación estándar

con un valor de 1,61 segundos de forma sistematizado lo que establece que la dispersión de datos es mínima en comparación a la dispersión de datos de forma tradicional representada por el valor de 3,64 segundos, también se obtuvo un tiempo mínimo de forma tradicional de 190 segundos y un máximo de 204 segundos, mientras que de forma sistematizada el mínimo fue de 63 segundos y el máximo de 69 segundos, como se puede visualizar en la **Tabla 2-3**.

Tabla 2-3: Estadística descriptiva del registro de pedidos

Tiempo en segundos del método Tradicional		Tiempo en segundos del método Sistematizado	
Media	196,3	Media	65,13
Desviación estándar	3,64	Desviación estándar	1,61
Mínimo	190	Mínimo	63
Máximo	204	Máximo	69

Realizado por: Adriana Quille, 2019.

3.1.2.2. Estadística Inferencial

Se aplicó la estadística inferencial con el fin de conocer si los resultados obtenidos con las muestras son aplicables para toda la población; para este análisis se utilizó la herramienta de Microsoft Excel y el software estadístico Minitab 15 para el análisis de datos.

- **Registro de pedidos**

1. Planteamiento de la hipótesis:

Hipótesis Nula (H_0) = El tiempo promedio sistematizado es igual al tiempo promedio tradicional para el registro de pedidos.

Hipótesis Alternativa (H_1) = El tiempo promedio sistematizado es diferente al tiempo promedio tradicional para el registro de pedidos.

2. Nivel de Significancia

Se trabajó con un nivel de significancia de $\alpha = 0.05$, dado el caso que es considerado como un margen de error mínimo garantizando así un 95% de confiabilidad.

3. Estadístico de Prueba

Se aplicó la *prueba t para medias de dos muestras emparejadas*, los datos evaluados anteriormente provienen de una distribución normal, el mismo que lo puede observar en el apartado **3.1.1.**, estos datos también son cuantitativos y menores a 30, permitiendo realizar la

comparativa entre los tiempos sistematizados vs el tradicional, a continuación, en la **Tabla 3-3**, se muestra los resultados obtenidos de la prueba estadística.

Tabla 3-3: Prueba T del registro de pedidos.

Prueba t para medias de dos muestras emparejadas	
Grados de libertad	29
Estadístico t	155,3563104
P(T<=t) dos colas	6,54505E-44

Realizado por: Adriana Quille, 2019.

4. Establecer regla de decisión

En el **Gráfico 1-3**, se puede visualizar la distribución *t-Student* para medias de dos muestras emparejadas, obteniendo el estadístico de prueba, la zona de aceptación y la zona de rechazo marcada de color rojo, la misma que se encuentra en las dos colas dado el caso que en la Hipótesis Alternativa se planteó comprobar si existe diferencia entre las dos medias.

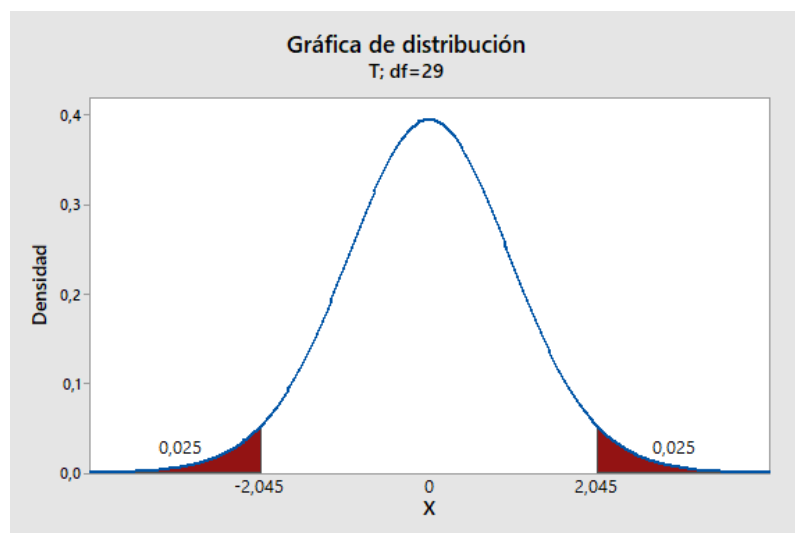


Gráfico 1-3: Gráfica de distribución T del registro de pedidos

Realizado por: Adriana Quille, 2019.

A continuación, en la **Tabla 4-3**, se puede observar los datos que se utilizó en la regla de decisión para el registro de pedidos, el t calculado (*tc*) es el valor estadístico t la misma que se puede visualizar en la **Tabla 3-3**, el t tratado (*tt*) es el valor que se puede observar en el **Gráfico 1-3**.

Tabla 4-3: Regla de decisión del registro de pedidos

T calculado (<i>tc</i>)	>	T tratado (<i>tt</i>)	
155,35	>	-2,045	Rechazo H_0
(p)	<	Nivel de Significancia(α)	

6,54505E-44	<	0,05	Rechazo H ₀
-------------	---	------	------------------------

Realizado por: Adriana Quille, 2019.

5. Toma de decisión

Dado el valor calculado t calculado (t_c) de 155,35, es mayor que el valor crítico de la cola izquierda (-2,045) rechazando la H_0 ; es decir que el tiempo promedio sistematizado es estadísticamente diferente al tiempo promedio tradicional del registro de pedidos con un nivel de significancia de 0,05; las diferencias de los tiempos de respuesta del registro de pedidos sistematizado y tradicional se muestran en el **Gráfico 2-3**.

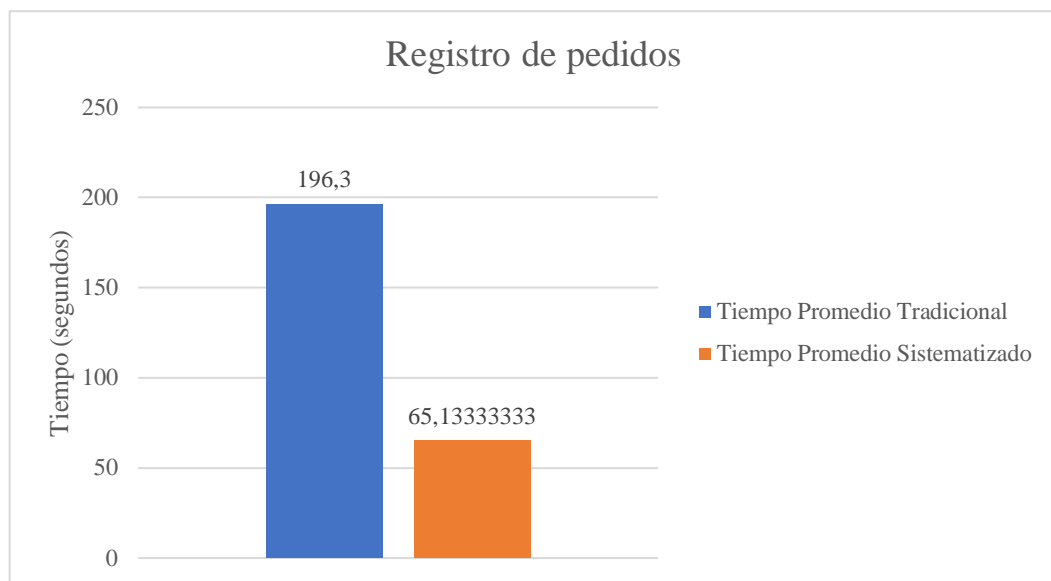


Gráfico 2-3: Tiempo promedio del registro de pedidos.

Realizado por: Adriana Quille, 2019.

En el **Gráfico 2-3**, se puede observar que para el registro de pedidos a través del sistema tradicional el tiempo promedio que se demora es de 196,3 segundos; mientras que aplicando el método sistematizado el tiempo promedio es de 65,13 segundos, por tanto, se dice que existe una disminución en el tiempo de ejecución de dicho proceso con la utilización del sistema web, el mismo que representa un 66,82% de reducción.

3.1.3. Caso 2

Este caso dos se encuentra dirigido a los trabajadores de la Fundación “Emprender y No Dependier”, se consideraron dos funcionalidades del sistema, para el rol “administrador” se

analizó el registro de datos de los emprendedores, y para el rol de “emprendedores” se analizó el registro de datos de los emprendimientos.

3.1.3.1. Estadística Descriptiva

Se aplicó la estadística descriptiva para analizar de forma general los datos que han sido recopilados, cabe mencionar que los datos analizados se encuentran en el **Anexo G**.

- **Registro de datos de los emprendimientos**

Para el registro de datos de los emprendimientos se obtuvo un total de 15 tiempos expresados en segundos, los mismos que permitieron evaluar la eficiencia del sistema, se tomó en cuenta la *media, desviación estándar, valor mínimo y máximo* de los tiempos obtenidos durante este proceso, en los cuales se obtuvo un tiempo promedio de 216,06 segundos en el método tradicional, mientras que con el método sistematizado se obtuvo un tiempo promedio de 122,46 segundos, la desviación estándar con un valor de 1,59 segundos de forma sistematizado lo que establece que la dispersión de datos es mínima en comparación a la dispersión de datos de forma tradicional que es mucho más alta representada por el valor de 16,90 segundos, también se obtuvo un tiempo mínimo de forma tradicional de 191 segundos y un máximo de 239 segundos, mientras que de formar sistematizada el mínimo fue de 120 segundos y el máximo de 125 segundos, como se puede visualizar en la **Tabla 5-3**.

Tabla 5-3: Estadística descriptiva del registro de datos de los emprendimientos

Tiempo en segundos del método Tradicional		Tiempo en segundos del método Sistematizado	
Media	216,06	Media	122,46
Desviación estándar	16,90	Desviación estándar	1,59
Mínimo	191	Mínimo	120
Máximo	239	Máximo	125

Realizado por: Adriana Quille, 2019.

3.1.3.2. Estadística Inferencial

Se aplicó la estadística inferencial con el fin de conocer si los resultados obtenidos con las muestras son aplicables para toda la población; para este análisis se utilizó la herramienta de Microsoft Excel y el software estadístico Minitab 15 para el análisis de datos.

- **Registro de datos de los emprendimientos**

1. Planteamiento de la hipótesis:

Hipótesis Nula (H_0) = El tiempo promedio sistematizado es igual al tiempo promedio tradicional para el registro de datos de los emprendimientos.

Hipótesis Alterna (H_1) = El tiempo promedio sistematizado es diferente al tiempo promedio tradicional para el registro de datos de los emprendimientos.

2. Nivel de Significancia

Se trabajó con un nivel de significancia de $\alpha = 0.05$, dado el caso que es considerado como un margen de error mínimo garantizando así un 95% de confiabilidad.

3. Estadístico de Prueba

Se aplicó la *prueba t para medias de dos muestras emparejadas*, los datos evaluados anteriormente provienen de una distribución normal, el mismo que lo puede observar en el apartado 3.1.1., estos datos también son cuantitativos y menores a 30, permitiendo realizar la comparativa entre los tiempos sistematizados vs el tradicional, a continuación, en la **Tabla 6-3**, se muestra los resultados obtenidos de la prueba estadística.

Tabla 6-3: Prueba T del registro de datos de los emprendimientos

Prueba t para medias de dos muestras emparejadas	
Grados de libertad	14
Estadístico t	21,41542437
P(T<=t) dos colas	4,24757E-12

Realizado por: Adriana Quille, 2019.

4. Establecer regla de decisión

En el **Gráfico 3-3**, se puede visualizar la distribución *t-Student* para medias de dos muestras emparejadas, obteniendo el estadístico de prueba, la zona de aceptación y la zona de rechazo marcada de color rojo, la misma que se encuentra en las dos colas dado el caso que en la Hipótesis Alternativa se planteó comprobar si existe diferencia entre las dos medias.

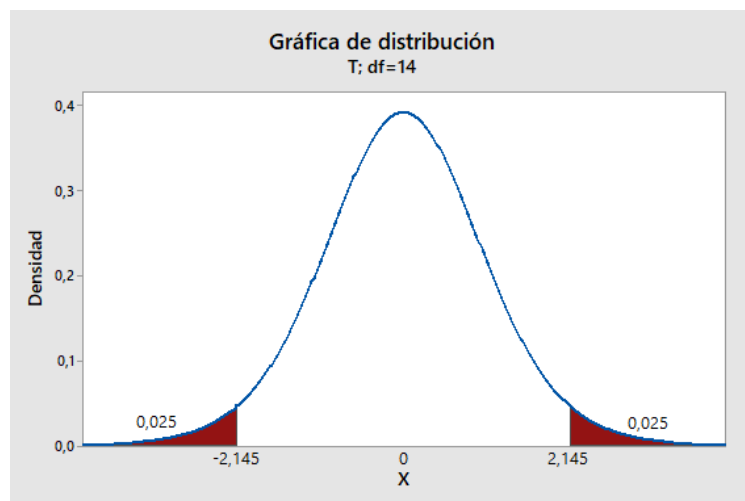


Gráfico 3-3: Gráfica de distribución T del registro de los emprendimientos
Realizado por: Adriana Quille, 2019.

A continuación, en la **Tabla 7-3**, se puede observar los datos que se utilizó en la regla de decisión para el registro de datos de los emprendimientos, el t calculado (t_c) es el valor estadístico t la misma que se puede visualizar en la **Tabla 6-3**, el t tratado (t_t) es el valor que se puede observar en el **Gráfico 3-3**.

Tabla 7-3: Regla de decisión del registro de datos de los emprendimientos

T calculado (t_c)	>	T tratado (t_t)	
21,41	>	-2,145	Rechazo H_0
(p)	<	Nivel de significancia (α)	
4,24757E-12	<	0,05	Rechazo H_0

Realizado por: Adriana Quille, 2019.

5. Toma de decisiones

Dado el valor t calculado (t_c) de 21,41, es mayor que el valor crítico de la cola izquierda (-2,145) rechazando la H_0 ; es decir que el tiempo promedio sistematizado es estadísticamente diferente al tiempo promedio tradicional del registro de datos de los emprendimientos con un nivel de significancia de 0,05; las diferencias de los tiempos de respuesta del registro de datos de los emprendimientos sistematizado y tradicional se muestran en el **Gráfico 4-3**.

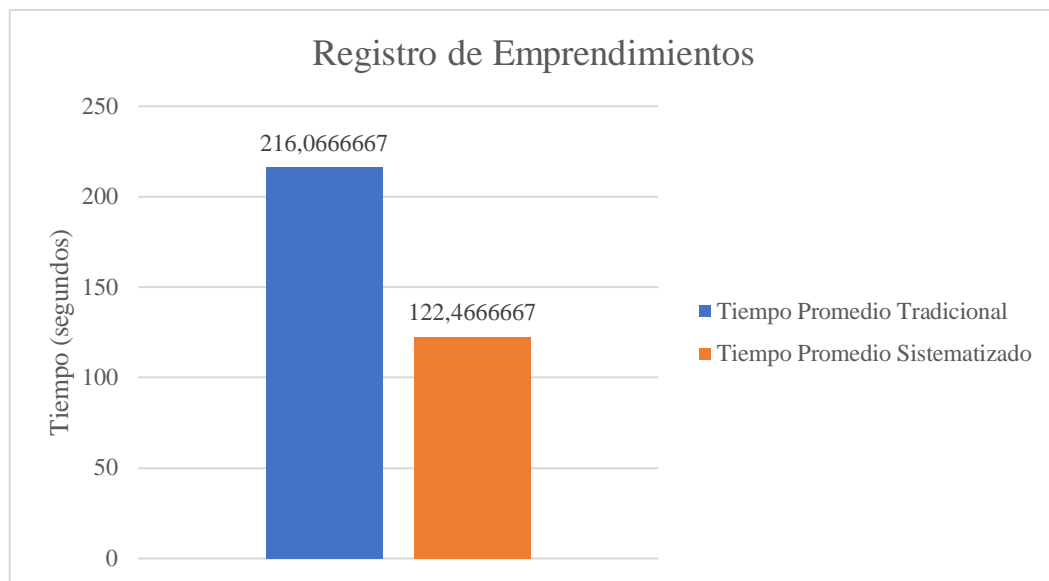


Gráfico 4-3: Tiempos promedios del Registro de datos de los emprendimientos.
Realizado por: Adriana Quille, 2019.

En el **Gráfico 4-3**, se puede observar que para el registro de datos de los emprendimientos a través del sistema tradicional el tiempo promedio que se demora es de 216,06 segundos; mientras que aplicando el método sistematizado el tiempo promedio es de 122,46 segundos, por tanto, se dice que existe una disminución en el tiempo de ejecución de dicho proceso con la utilización del sistema web, el mismo que representa un 56,68% de reducción.

3.1.3.3. Estadística Descriptiva

Se aplicó la estadística descriptiva para analizar de forma general los datos que han sido recopilados, cabe mencionar que los datos analizados se encuentran en el **Anexo G**.

- **Registro de datos de los emprendedores**

Para el registro de datos de los emprendedores se obtuvo un total de 10 tiempos expresados en segundos, los mismos que permitieron evaluar la eficiencia del sistema, se tomó en cuenta la *media*, *desviación estándar*, *valor mínimo* y *máximo* de los tiempos obtenidos durante este proceso, en los cuales se obtuvo un tiempo promedio de 150,4 segundos en el método tradicional, mientras que con el método sistematizado se obtuvo un tiempo promedio de 84,3 segundos, la desviación estándar con un valor de 13,08 segundos de forma sistematizado lo que establece que la dispersión de datos es mínima en comparación a la dispersión de datos de forma tradicional que es mucho más alta representada por el valor de 23,44 segundos, también se obtuvo un tiempo mínimo de forma tradicional de 120 segundos y un máximo de 179 segundos, mientras que de

formar sistematizada el mínimo fue de 62 segundos y el máximo de 104 segundos, como se puede visualizar en la **Tabla 8-3**.

Tabla 8-3: Estadística descriptiva del registro de datos de los emprendedores

Tiempo en segundos del método Tradicional		Tiempo en segundos del método Sistematizado	
Media	150,4	Media	84,3
Desviación estándar	23,44	Desviación estándar	13,08
Mínimo	120	Mínimo	62
Máximo	179	Máximo	104

Realizado por: Adriana Quille, 2019.

3.1.3.4. Estadística Inferencial

Se aplicó la estadística inferencial con el fin de conocer si los resultados obtenidos con las muestras son aplicables para toda la población; para este análisis se utilizó la herramienta de Microsoft Excel y el software estadístico Minitab 15 para el análisis de datos.

- **Registro de datos de los emprendedores**

1. Planteamiento de la hipótesis:

Hipótesis Nula (H_0) = El tiempo promedio sistematizado es igual al tiempo promedio tradicional para el registro de datos de los emprendedores.

Hipótesis Alterna (H_1) = El tiempo promedio sistematizado es menor al tiempo promedio tradicional para el registro de datos de los emprendedores.

2. Nivel de Significancia

Se trabajó con un nivel de significancia de $\alpha = 0.05$, dado el caso que es considerado como un margen de error mínimo garantizando así un 95% de confiabilidad.

3. Estadístico de Prueba

Se aplicó la *prueba t para medias de dos muestras emparejadas*, los datos evaluados anteriormente provienen de una distribución normal, el mismo que lo puede observar en el apartado **3.1.1.**, estos datos también son cuantitativos y menores a 30, permitiendo realizar la comparativa entre los tiempos sistematizados vs el tradicional, a continuación, en la **Tabla 9-3**, se muestra los resultados obtenidos de la prueba estadística.

Tabla 9-3: Prueba T del registro de datos de los emprendedores

Prueba t para medias de dos muestras emparejadas	
Grados de libertad	9
Estadístico t	7,145818354
P(T<=t) dos colas	5,39083E-05

Realizado por: Adriana Quille, 2019.

4. Establecer regla de decisión

En el **Gráfico 5-3**, se puede visualizar la distribución *t-Student* para medias de dos muestras emparejadas, obteniendo el estadístico de prueba, la zona de aceptación y la zona de rechazo marcada de color rojo, la misma que se encuentra en las dos colas dado el caso que en la Hipótesis Alternativa se planteó comprobar si existe diferencia entre las dos medias.

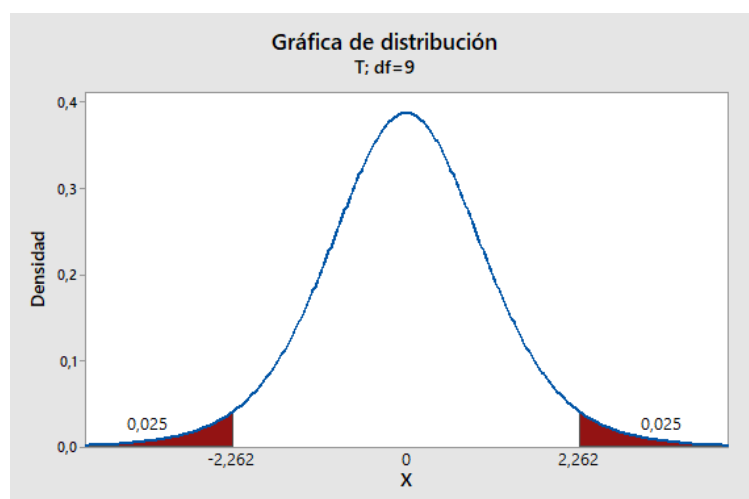


Gráfico 5-3: Gráfica de distribución T del registro del emprendedor.
Realizado por: Adriana Quille, 2019.

A continuación, en la **Tabla 10-3**, se puede observar los datos que se utilizó en la regla de decisión para el registro de datos de los emprendedores, el t calculado (*tc*) es el valor estadístico t la misma que se puede visualizar en la **Tabla 9-3**, el t tratado (*tt*) es el valor que se puede observar en el **Gráfico 5-3**.

Tabla 10-3: Regla de decisión del registro de datos de los emprendedores

T calculado (<i>tc</i>)	>	T tratado (<i>tt</i>)	
7,14	>	-2,262	Rechazo H_0
(p)	<	(α)	
5,39083E-05	<	0,05	Rechazo H_0

Realizado por: Adriana Quille, 2019.

5. Toma de decisiones

Dado el valor t calculado (t_c) de 7,14, es mayor que el valor crítico de la cola izquierda (-2,262) rechazando la H_0 ; es decir que el tiempo promedio sistematizado es estadísticamente diferente al tiempo promedio tradicional del registro de datos de los emprendedores con un nivel de significancia de 0,05; las diferencias de los tiempos de respuesta del registro de datos de los emprendedores sistematizado y tradicional se muestran en el **Gráfico 6-3**.

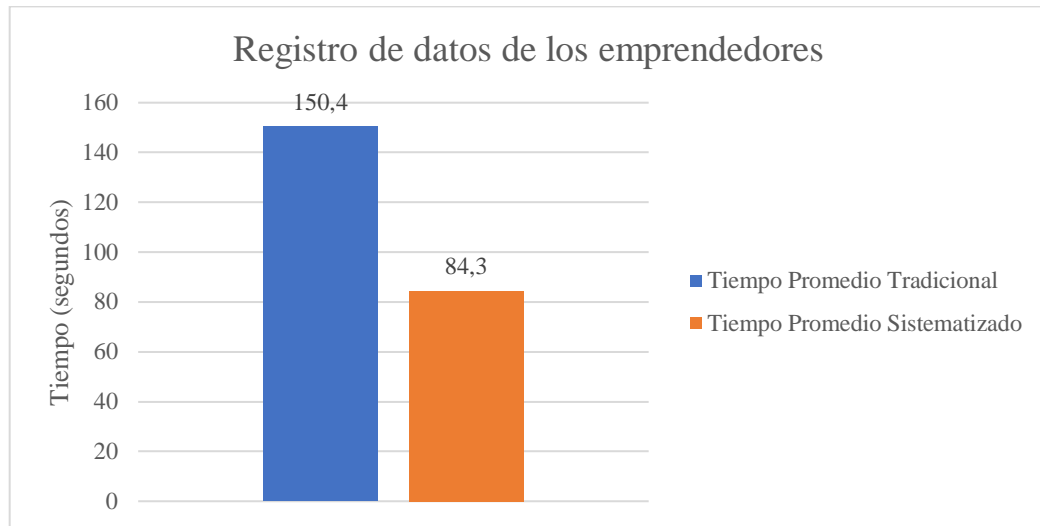


Gráfico 6-3: Tiempos promedios del registro de datos de los emprendedores
Realizado por: Adriana Quille, 2019.

En el **Gráfico 6-3**, se puede observar que para el registro de datos de los emprendimientos a través del sistema tradicional el tiempo promedio que se demora es de 150,4 segundos; mientras que aplicando el método sistematizado el tiempo promedio es de 84,3 segundos, por tanto, se dice que existe una disminución en el tiempo de ejecución de dicho proceso con la utilización del sistema web, el mismo que representa un 56,05% de reducción.

Una vez terminado la evaluación del sistema web en base al comportamiento de tiempos se demuestra que el tiempo promedio sistematizado es estadísticamente diferente al tiempo promedio tradicional para los procesos analizados con un nivel de significancia del 0,05% y además obteniendo una reducción entre el 56,05% hasta 66,82% con un promedio de 59,85% de mejora de la eficiencia de los procesos analizados.

CONCLUSIONES

- Del análisis realizado en la Fundación “Emprender y No Dependier” con los indicadores de los emprendimientos y la gestión de pedidos se determinó 20 actividades que contemplan estos dos procesos para la gestión y difusión de los emprendimientos en la Fundación “Emprender y No Dependier”.
- Del estudio de los frameworks por lado del servidor (BackEnd) se determinó que Symfony es el adecuado para el desarrollo del sistema de la Fundación “Emprender y No Dependier”, el mismo que incorpora el patrón de diseño MVC y es orientado a objetos permitiendo obtener un código ordenado.
- En el desarrollo del chat en línea del sistema web se analizó las tecnologías PHPWebSocket, Socket.IO y jWebSocket, concluyendo que PHPWebSocket es la mejor porque soluciona los errores de codificación y personalización de acuerdo con las necesidades del desarrollador.
- La aplicación de la metodología SCRUM permitió controlar los tiempos de ejecución con los tiempos estimados, teniendo un desfase en el sprint 5 que fue compensado en el sprint 9 dado que se sobreestimó las tareas planificadas, cabe recalcar que el desarrollo del sistema web no se vio afectado ante estos inconvenientes.
- En el desarrollo del sistema web se utilizó el framework Symfony, para encapsular los gestores del sistema (ingresar, modificar, eliminar, buscar), el framework Bootstrap para la adaptabilidad del sistema web a cualquier dispositivo y la tecnología WebSocket para la comunicación bidireccional del chat entre el cliente y administrador.
- Se utilizó la metodología ágil SCRUM para el desarrollo del sistema web, la misma que tuvo una duración de 960 horas lo que implica que se desarrolló 41 requerimientos funcionales, cada uno de estos se encuentran documentado con un total de 182 tareas de ingeniería, 276 pruebas de aceptación y 8 historias técnicas resultando todas estas como exitosas.
- El sistema web de la Fundación “Emprender y No Dependier” se desarrolló con el patrón MVC permitiendo obtener un sistema flexible, dinámico y con una estructura modular.
- Con la implementación del sistema web en la Fundación “Emprender y No Dependier” se concluye que la eficiencia de los procesos analizados mejoró en un promedio de 59,85% con un nivel de significancia del 0,05.

RECOMENDACIONES

Se recomienda:

- Utilizar las herramientas Symfony, Bootstrap y MySQL para el desarrollo de los sistemas web debido a que son fáciles de aprender y permite obtener resultados positivos ante cualquier dificultad que se presente.
- Descargar desde la página oficial de Symfony y mantener actualizado este framework para optimizar y mejorar su funcionamiento.
- Evaluar la eficiencia de los sistemas web aplicando la norma ISO/IEC 25000 por ser la más actualizada que fue reemplazó a la ISO/IEC 9126.
- Desarrollar el módulo funcional gestión de ventas de los pedidos realizados para completar el sistema web y el módulo de notificaciones para el chat en línea para contar con un historial del chat remitente.

BIBLIOGRAFÍA

ACHOUR, M; et al. BRIONES, G. *Manual de PHP* [en línea]. S.l.: 8 Junio, 2003. [Consulta: 14 enero 2019]. Disponible en: <http://www.opencontent.org/openpub/>.

ACOSTA, M; et al. RODALLEGA, C. *Symfony*. [en línea]. S.l.: 13 Septiembre, 2014. [Consulta: 1 mayo 2019]. Disponible en: <https://slideplayer.es/slide/94244/>.

ANDINO, C. Desarrollo de un sistema académico orientado a la web para la unidad educativa juan de velasco utilizando symfony y mysql [en línea] (Tesis) (PreGrado) Escuela Superior Politécnica de Chimborazo, Chimborazo, Ecuador, 2016. [Consulta: 14 enero 2019]. Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/220/1/236T0001.pdf>.

ALLEGUE, A. *Comparación de metodologías: Agile-Scrum vs Tradicional - Lecciones Aprendidas*. [blog]. Switzerland: 4 Septiembre, 2018. [Consulta: 1 mayo 2019]. Disponible en: <https://lecciondeaprendizaje.blogspot.com/2018/03/comparacion-de-metodologias-agile-scrum.html>.

ARCOS, G; et al. VALLEJO, J., "Comparative Study of Performance and Productivity of MVC and MVVM design patterns". *KnE Engineering* [en línea], 2018, (Ecuador) 1(2), pp. 241-250. [Consulta: 14 enero 2019]. DOI 10.18502/keg.v1i2.1498. Disponible en: <https://knepublishing.com/index.php/KnE-Engineering/article/view/1498>.

BAEZ, S. *Sistemas Web :: KnowDo* [en línea]. Noruega: 15 Febrero, 2018. [Consulta: 5 junio 2019]. Disponible en: <http://www.knowdo.org/knowledge/39-sistemas-web>.

BAHIT, E. *POO y MVC en PHP*. [blog]. [Consulta: 14 enero 2019]. Disponible en: <http://eugeniabahit.blogspot.com>.

BASCÓN, E. *El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing*. [en línea]. Diciembre (Buenos Aires), 2014. 2(4) [Consulta: 14 enero 2019]. Disponible en: <https://www.mendeley.com/catalogue/el-patr%C3%B3n-dise%C3%B1o-modelovistacontrolador-mvc-y-su-implementaci%C3%B3n-en-java-swing/>

BBVAOPEN4U. *Ventajas e inconvenientes de Python y R para la ciencia de datos* [en línea]. España: 9 agosto, 2016. Disponible en: <https://bbvaopen4u.com/es/actualidad/ventajas-e-inconvenientes-de-python-y-r-para-la-ciencia-de-datos>.

BHARAT. *Advantages of Bootstrap framework | Vmoksha* [en línea]. 27 Agosto, 2018. [Consulta: 22 enero 2019]. Disponible en: <https://vmokshagroup.com/blog/bootstrap-advantages/>.

BUSTOS, G. *¿Qué es Apache? Descripción completa del servidor web Apache* [en línea]. 18 Enero, 2019. [Consulta: 30 abril 2019]. Disponible en: <https://www.hostinger.es/tutoriales/que-es-apache/>.

BYSSLENDER. *Características de la arquitectura Cliente-Servidor – Programación 7 ULAT sede Azuero* [en línea]. 24 Mayo, 2016. [Consulta: 8 enero 2019]. Disponible en: <https://programacionviiulatbysslender.wordpress.com/2016/05/24/caracteristicas-de-la-arquitectura-cliente-servidor/>.

CAKE SOFTWARE FOUNDATION. *CakePHP Cookbook Documentation* [en línea]. Washington-USA.: CakePHP, 12 Junio, 2018. [Consulta: 7 junio 2019]. Disponible en: https://book.cakephp.org/3.0/_downloads/es/CakePHPCookbook.pdf.

CAMACHO, G; et al. DANIEL, A. "Scientia et technica" [en línea]. Colombia: vol 2, nº44 (1995), (Universidad Tecnológica de Pereira) pp. 503-511. [Consulta: 22 enero 2019]. Disponible en: <https://www.redalyc.org/html/849/84917316032/>.

CAÑAVATE, A.M. "Los sistemas de información web como elementos de difusión de información al ciudadano. análisis y evolución de las prestaciones ofrecidas" [en línea], 2014, (España - Portugal) 1(1), pp. 2-5. [Consulta: 4 enero 2019]. Disponible en: http://www.iskoiberico.org/wp-content/uploads/2014/09/277-287_Muñoz.pdf.

CARVAJAL, M. "Sistema de información geográfica para la gestión de parqueaderos privados utilizando el framework phonegap y metodología de desarrollo mobile-d" [en línea], 2016, (Ibarra - Ecuador). [Consulta: 7 junio 2019]. Disponible en: http://repositorio.utn.edu.ec/bitstream/123456789/7805/5/04_ISC_396_Articulo.pdf.

CERVANTES, H. *Arquitectura de Software | SG Buzz* [en línea]. 7 Mayo, 2018. [Consulta: 9 junio 2019]. Disponible en: <https://sg.com.mx/revista/27/arquitectura-software>.

CLAROS, E. *INFORME FINAL DE INVESTIGACIÓN "Desarrollo de una Plataforma Web para la Comunicación y Difusión del Emprendimiento en el Ecuador"* [en línea]. El Salvador: Febrero, 2017. [Consulta: 4 enero 2019]. ISBN 978-99961-50-53-1. Disponible en: <https://www.itca.edu.sv/wp-content/uploads/2018/03/04-LU-Ecosistema-emprendedor.pdf>.

CUBILLOS, C. *Arquitectura Cliente/Servidor* [en línea]. Febrero, 2015. [Consulta: 8 enero 2019]. Disponible en: <http://ocw.pucv.cl/cursos-1/arquitectura-de-sistemas-de-software/materiales-de-clases/web-cliente-servidor>.

DESIRE. *Scrum – DESIRE* [en línea]. 12 Julio, 2017. [Consulta: 4 mayo 2019]. Disponible en: <https://desire.webs.uvigo.es/contenidos/scrum/>.

DÍAZ, M. *Motivos para que tu empresa tenga una aplicación web o de escritorio* [en línea]. 4

Mayo, 2017. [Consulta: 5 enero 2019]. Disponible en: <https://www.fuegoyamana.com/blog/aplicacion-web-o-de-escritorio-para-tu-negocio/>.

¿Es conveniente utilizar frameworks para desarrollo front-end? | 4R Soluciones | Diseño, Desarrollo y Programación Web & Mobile [blog]. 13 Noviembre, 2013. [Consulta: 4 enero 2019]. Disponible en: <http://www.4rsoluciones.com/blog/es-conveniente-utilizar-frameworks-para-desarrollo-front-end-2/>.

DURÁN, D. *ProQuest Ebook Central Reader* [en línea]. Cueva de Viera - España: 2015 1(1). [Consulta: 23 enero 2019]. Disponible en: <https://ebookcentral.proquest.com/lib/espoehsp/reader.action?docID=5214364&ppg=1>.

ECURED. *Arquitectura Cliente Servidor - EcuRed* [en línea]. Marzo, 2016. [Consulta: 8 enero 2019]. Disponible en: https://www.ecured.cu/Arquitectura_Cliente_Servidor.

EGUILUZ, J. *Symfony 4, Symfony Flex y el futuro de Symfony* [en línea]. Castellón: 1 Julio, 2017. [Consulta: 1 mayo 2019]. Disponible en: <https://www.slideshare.net/javier.eguiluz/desympfony-2017-symfony-4-symfony-flex-y-el-futuro-de-symfony>.

ENRÍQUEZ, A; et al. NOGUERON, G. *MySQL ¿Qué es MySQL?* [en línea]. 15 Junio, 2019. [Consulta: 23 enero 2019]. Disponible en: <http://www.gridmorelos.uaem.mx/~mcruz/cursos/miic/MySQL.pdf>.

ESPINOZA, E. *Ventajas y desventajas de los servidores apache y IIS* [en línea]. 25 Mayo, 2011. [Consulta: 30 abril 2019]. Disponible en: <https://es.slideshare.net/elianaespinoza/ventajas-y-desventajas-de-los-servidores-apache-y1>.

FLORES, A. *Un Análisis Regional de la Arquitectura Cliente / Servidor-Edición Única* [en línea]. (Tesis)(Maestría) INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY. California, Agosto, 2018. [Consulta: 23 enero 2019]. Disponible en: <https://www.mendeley.com/catalogue/un-an%C3%A1lisis-regional-la-arquitectura-cliente-servidoredici%C3%B3n-%C3%BAnica/>

GARCÍA, J. *¿Qué es Ruby? | OpenWebinars* [blog]. 20 Octubre, 2017. [Consulta: 9 enero 2019]. Disponible en: <https://openwebinars.net/blog/que-es-ruby/>.

GÓMEZ, A. *Desarrollo de un sistema web: elementos, front-end y backend. — Steemit* [en línea]. 3 Febrero, 2018. [Consulta: 4 enero 2019]. Disponible en: <https://steemit.com/spanish/@angelggomz/desarrollo-de-un-sistema-web-elementos-front-end-y-backend>.

GOMEZ, K. *Top 5 Metodologías de Desarrollo de Software* [en línea]. 27 Julio, 2017. [Consulta:

30 abril 2019]. Disponible en: <https://www.megapractical.com/blog-de-arquitectura-soa-y-desarrollo-de-software/metodologias-de-desarrollo-de-software>.

GÓMEZ, V. *Arquitectura en Tres Capas - Instinto Binario* [en línea]. 23 Junio, 2015. [Consulta: 9 junio 2019]. Disponible en: <https://instintobinario.com/arquitectura-en-tres-capas/>.

GONÇALVES, L. *Qué es la metodología Ágil* [en línea]. 25 Enero, 2019. [Consulta: 4 junio 2019]. Disponible en: <https://luis-goncalves.com/es/que-es-la-metodologia-agil/>.

GONZÁLEZ, J. *Desarrollo de sitios web con PHP y MySQL Tema 1: Introducción* [en línea]. S.l.: 2015. [Consulta: 9 enero 2019]. Disponible en: <http://www.lsi.us.es/cursos/cursophp/apuntes/tema1.pdf>.

GONZALEZ, M. *Arquitectura cliente servidor* [en línea]. S.l.: 3 Julio, 2018. [Consulta: 9 junio 2019]. Disponible en: <https://es.slideshare.net/ester.gonzalez/arquitectura-cliente-servidor-104004600>.

GRÁFICA, R. *El lenguaje de programación PHP - Red Gráfica Latinoamérica* [en línea]. S.l.: 3 Marzo, 2016. [Consulta: 10 enero 2019]. Disponible en: <http://redgrafica.com/El-lenguaje-de-programacion-PHP>.

GRUPO, E. *Chat Online para tu sitio web* [en línea]. 3 Noviemrbe 2018. [Consulta: 26 enero 2019]. Disponible en: <https://www.grupoclipse.net/chat-online/index.html>.

HANSA, U. *Comienza Hoy a usar WebSockets de HTML5* [en línea]. 28 Julio 2016. [Consulta: 6 junio 2019]. Disponible en: <https://code.tutsplus.com/es/tutorials/start-using-html5-websockets-today--net-13270>.

HMD PROJECT MANAGERS. *Cuáles son las reglas del KANBAN, algunos tips y reflexiones* [en línea]. 17 Abril, 2016. [Consulta: 4 junio 2019]. Disponible en: <https://uv-mdap.com/blog/cuales-son-las-reglas-del-kanban-algunos-tips-y-reflexiones/>.

ALBERTOS GÓMEZ, E; et al. DÍAZ FERNANDEZ, J. *Arquitecturas Software para Microservicios: Una Revisión Sistemática de la Literatura* Tutor: Co-Tutor. [en línea] (Tesis) (Máster) Universidad Politécnica de Madrid, España, 2018. pp. 5-8. [Consulta: 9 junio 2019]. Disponible en: http://oa.upm.es/51460/1/TESIS_MASTER_ELENA_ALBERTOS_GOMEZ.pdf.

LAFOSSE, J. *Struts 2 : el framework de desarrollo de aplicaciones Java EE*. S.l.: Ediciones Eni. ISBN 9782746055421. Abril, 2010. [Consulta: 22 enero 2019]. Disponible en: <https://www.mendeley.com/catalogue/un-an%C3%framework-desarrollo-dici%C3%B3n-%C3%JavaEE/>

LEMUS OLALDE, C. *Calidad de Software: Modelos, Procesos, Arquitectura* [en línea]. S.l.: Septiembre, 2007. [Consulta: 23 abril 2019]. Disponible en: <https://www.cimat.mx/Eventos/seminariodetecnologias/handout-CLemus.pdf>.

LEÓN, Á.R.E; et al. MAYORGA, I.I.C. "Patrón MVC, un componente para la implementación de una Estrategia Informática para mejorar gestión de datos en el área de estadística: Caso de Estudio Hospital Maternidad Babahoyo." [en línea], 2016, (Universidad Regional Autónoma de los Andes) 3(4), pp. 4-5. [Consulta: 14 enero 2019]. Disponible en: <http://45.238.216.13/ojs/index.php/EPISTEME/article/view/280/240>.

LIZAMA; et al. JERIA, J. "Redes de computadores Arquitectura Cliente-Servidor" [en línea], 2016, (Universidad Técnica Federico Santa María) 1(2), p. 4. [Consulta: 8 enero 2019]. Disponible en: <http://profesores.elo.utfsm.cl/~agv/elo322/1s16/projects/reports/Proyecto Cliente - Servidor.pdf>.

LÓPEZ GIL, A. Estudio comparativo de metodologías tradicionales y ágiles para proyectos de Desarrollo de Software [en línea] (Tesis) (Pre Grado) Universidad de Valladolid, España, 2018. pp. 26-45. [Consulta: 1 mayo 2019]. Disponible en: <http://uvadoc.uva.es/bitstream/10324/32875/1/TFG-I-1015.pdf>.

M.I.D SALAZAR, L. 4.2. *Arquitectura cliente/servidor* [en línea]. Septiembre, 2016. [Consulta: 8 enero 2019]. Disponible en: http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro21/42_arquitectura_clienteservidor.html.

MALDONADO GUERRERO, José Rodrigo. Desarrollo e implementación de un sistema web de seguimiento y evaluación de las prácticas pre-profesionales para la facultad de ingeniería escuela civil de la puce. [en línea]. (Tesis) (Pregrado). PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR FACULTAD DE INGENIERÍA ESCUELA CIVIL. Quito, Ecuador, 2016. pp. 1-198. [Consulta: 8 enero 2019]. Disponible en: http://dspace.puce.edu.ec/bitstream/sistematicivil/123456789/9117/5/Tesis_Teoria.pdf.

MARABOLI, M. *DE UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA CERTIFICADA EN INFORMACIÓN SECURITY SPECIALIST PROFESSIONAL (CISSP)*. S.l.: s.n. ISBN 6666666666, 2015. [Consulta: 8 enero 2019]. Disponible en: <https://www.mendeley.com/catalogue/manual-programaci%C3%B3n-en-php/>

MARTINEZ, S. *¿Qué es la metodología ágil? - Superrhheroes* [en línea]. 23 Noviembre 2018. [Consulta: 4 junio 2019]. Disponible en: <https://superrhheroes.sesametime.com/la-metodologia-agil/>.

MATSUSHITA; et al. PHP総合研究所. "Tecnologías de Comunicación Síncrona aplicadas al Sistema de Información para Tutoría: Acerkte" [en línea], 2016, (Instituto Tecnológico Superior de Comalcalco, México) 5(10), p. 3. [Consulta: 26 enero 2019]. Disponible en: <http://www.reci.org.mx/index.php/reci/article/view/45/204>.

MICROSOFT. *Capítulo 5. Cliente-Servidor. 5.1 Introducción* [en línea]. 1(2), 2016, (Microsoft Word). pp. 2-4. [Consulta: 8 enero 2019]. Disponible en: http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm/capitulo5.pdf.

MICROSOFT. *SQL Server 2017 en Windows y Linux / Microsoft* [en línea]. 2018. [Consulta: 22 enero 2019]. Disponible en: <https://www.microsoft.com/es-es/sql-server/sql-server-2017>.

MICROSOFT. *Conceptos básicos sobre bases de datos - Access* [en línea]. Brasil: 5 Noviembre, 2018. [Consulta: 22 enero 2019]. Disponible en: <https://support.office.com/es-es/article/conceptos-básicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204>.

MIRÓ, A. *Características PHP* [en línea], 27 Enero, 2016. [Consulta: 22 enero 2019]. Disponible en: <https://www.deustoformacion.com/blog/programacion-diseno-web/7-caracteristicas-lenguaje-php-que-lo-convierten-uno-mas-potentes>.

MYSQL. *MySQL :: MySQL 5.0 Reference Manual :: 1.4.2 Las principales características de MySQL* [en línea]. 2018. [Consulta: 23 enero 2019]. Disponible en: <http://ftp.tcrc.edu.tw/MySQL/doc/refman/5.0/es/features.html>.

NAVARRETE, E.N. & NAVARRO, R. *Desarrollo de un marco de trabajo (framework) para el desarrollo de aplicaciones web en la Universidad Nacional de Costa Rica* [en línea]. (Tesis (Pregrado), UNIVERSIDAD NACIONAL DE COSTA RICA, 2017. pp. 12. [Consulta: 7 junio 2019]. Disponible en: [http://documentas.redclara.net/bitstream/10786/1287/1/131-17-4Desarrollo de un marco de trabajo %28framework%29 para el desarrollo de aplicaciones web.pdf](http://documentas.redclara.net/bitstream/10786/1287/1/131-17-4Desarrollo%20de%20un%20marco%20de%20trabajo%20framework%20para%20el%20desarrollo%20de%20aplicaciones%20web.pdf).

NEOSOFT. *¿Qué es una aplicación Web? – Blog Neosoft Sistemas* [en línea]. 8 Enero, 2018. [Consulta: 5 enero 2019]. Disponible en: <https://www.neosoft.es/blog/que-es-una-aplicacion-web/>.

ORACLE. *MySQL :: MySQL 8.0 Reference Manual :: 1 Información general* [en línea]. Diciembre, 2018. [Consulta: 22 enero 2019]. Disponible en: <https://dev.mysql.com/doc/refman/8.0/en/introduction.html>.

PANADERO, C. *Arquitectura de software y sus beneficios - DevExperto* [en línea]. 11 Febrero, 2017. [Consulta: 8 junio 2019]. Disponible en: <https://devexperto.com/arquitectura-del-software/>.

PÉREZ, C; et al. CASILLAS, A. *Bases de datos 71Z799014MO* [en línea]. 23 Abril, 2018. [Consulta: 22 enero 2019]. Disponible en: www.glo.org.mx.

PÉREZ, D. *Los diferentes lenguajes de programación para la web* [en línea]. 2 Noviembre, 2017. [Consulta: 9 enero 2019]. Disponible en: <http://www.maestrosdelweb.com/los-diferentes-lenguajes-de-programacion-para-la-web/>.

PERU, W.S. *Sistemas Web, Aplicación web, Sistema Web, Desarrollo sistema Perú.* [en línea]. Septiembre, 2018. [Consulta: 4 enero 2019]. Disponible en: <https://websystemperu.com/sistemas-web>.

PICUCCI, M; et al. CECCHI, L. *Entorno Web para la Programación del Robot Frankestito.* S.l.: ISBN 978-987-3619-27-4. 2018. [Consulta: 7 enero 2019]. Disponible en: <https://www.mendeley.com/catalogue/entorno-web-para-programaci%C3%B3n-en-php/>

POTENCIER, F. & ZANINOTTO, F. *Symfony la guía definitiva* [en línea]. S.l.: 2018. [Consulta: 14 enero 2019]. Disponible en: www.librosweb.es.

POTINENI, P; et al. YAGOUB, K. *Oracle® Database 2 Day DBA 18c* [en línea]. España: Scholarly, 2018. [Consulta: 22 enero 2019]. Disponible en: <https://docs.oracle.com/en/database/oracle/oracle-database/18/admqsl/2-day-dba.pdf>.

QUIZHPE, V; et al. OLIVO, A. "Factores que inciden en la sostenibilidad de los emprendimientos en el cantón las naves provincia bolívar 2015-2016.", (2017). pp. 1-20. [Consulta: 26 enero 2019]. Disponible en: <https://www.mendeley.com/catalogue/factores-que-inciden-en-la-sostenibilidad-los-emprendimientos-en-el-cant%C3%B3n-las-naves-provincia-bo%C3%ADv/>

ROA, A. *WebSocket y PHP Socket – Alejandro Roa – Medium* [en línea]. 15 Diciembre, 2016. [Consulta: 26 enero 2019]. Disponible en: <https://medium.com/@alejandrora/websocket-y-php-socket-3d8ac636a108>.

SAAVEDRA, H. *CARACTERÍSTICAS ARQUITECTURA CLIENTE-SERVIDOR – PROGRAMACION 7 UNIVERSIDAD LATINA SEDE AZUERO.* [en línea]. 24 Mayo, 2016. [Consulta: 8 enero 2019]. Disponible en: <https://programacion7ulatsaavas.wordpress.com/2016/05/24/caracteristicas-arquitectura-cliente-servidor/>.

SADASIVAN, G; et al. QUITTEK, J. *Architecture for IP flow information export* [en línea]. S.l.: RFC Editor, Diciembre, 2011. [Consulta: 8 junio 2019]. Disponible en: <https://tools.ietf.org/html/rfc6455>.

SALAZAR, CECILIA. "Diseño e implementaación de una plataforma web de E-Turismo en la

oficina de gestión y promoción turística de Cotopaxi" [en línea]. (Tesis) (Maestría) Universidad de las Fuerzas Armadas, Latacunga-Cotopaxi, 2015. pp. 13-15. [Consulta: 23 enero 2019]. Disponible en: <http://repositorio.espe.edu.ec/bitstream/21000/10128/1/T-ESPEL-MAS-0020.pdf>.

SÁNCHEZ, C. & CANO, N. "Introducción A La Programación" [en línea], 2014 (Chía - Cundinamarca), 1(2). p. 3. [Consulta: 6 enero 2019]. Disponible en: http://files.sanjo2014.webnode.es/200000001-c34cac445e/INTRODUCCION_A_LA_PROGRAMACION.pdf.

SMARTSYS-ISO 9126. *Norma Iso-9126 para análisis de software / Smartsys* [en línea]. Marzo, 2015. [Consulta: 23 abril 2019]. Disponible en: <http://www.smartsys.com.ec/?p=391>.

SOCKET. *Socket.IO* [en línea]. 12 Mayo, 2019. [Consulta: 8 junio 2019]. Disponible en: <https://socket.io/>.

STUDENTPLACE. *Metodología de Desarrollo de Software* [en línea]. 9 Septiembre, 2018. [Consulta: 30 abril 2019]. Disponible en: <https://studentplace98.blogspot.com/2018/09/metodologia-de-desarrollo-de-software.html>.

STUDIO, D. Web Illusion. *¿Qué es un framework web y qué ventajas aportan?- Diseño web Valencia* [en línea]. 25 Abril, 2018. [Consulta: 4 enero 2019]. Disponible en: <https://www.illusionstudio.es/que-es-un-framework-web>.

SUAREZ, P. *Usar o no un Framework - Kabytes* [en línea]. 6 Febrero, 2015. [Consulta: 20 febrero 2019]. Disponible en: <https://www.kabytes.com/programacion/usar-o-no-un-framework/>.

SYMPHONY. *¿Qué es Symfony?* [en línea]. Enero, 2019. [Consulta: 7 junio 2019]. Disponible en: <https://symfony.es/pagina/que-es-symfony/>.

TECHNOLOGIES, Z. *Zend Framework 2 Documentation* [en línea]. España: 2017. [Consulta: 7 junio 2019]. Disponible en: <https://buildmedia.readthedocs.org/media/pdf/zf2/latest/zf2.pdf>.

TODOPOSTGRESQL. *Ventajas y Desventajas de PostgreSQL - TodoPostgreSQL.* [en línea]. 30 Agosto, 2018. [Consulta: 22 enero 2019]. Disponible en: <https://todopostgresql.com/ventajas-y-desventajas-de-postgresql/>.

TRUJILLO, O. *Lenguajes de programación. ¿qué son y para que sirven? – MODS* [blog]. 18 Mayo, 2017. [Consulta: 6 enero 2019]. Disponible en: <http://www.mods.la/blog/lenguajes-de-programacion-que-son-y-para-que-sirven/>.

TUPE, C. & CISNEROS, J. *Evaluación y Selección de Framework de Desarrollo PHP: Symfony, Kumbia, CakePHP y Zend* [en línea]. España: Junio, 2008. [Consulta: 14 enero 2019].

Disponible en: https://2d34ee25-a-62cb3a1a-sites.googlegroups.com/site/laleydelacarretera/Home/EvaluaciónySeleccióndeFrameworkDesarrolloPHP-CakePhp%2CZend%2CKumbiaySymfony.pdf?attachauth=ANoY7cpv8gukZzorbydzAQ5jDDpARrFSSdoBiGvFyhIThGI2_RewlheKHYS6TF5ERIBDoJee9g-h-hZzN3MI8r5IHqtUTGqemBBtm4dlq9scjg_7e8WZ3caUaqJRFj_r78oM0t3WIP2_yrF2ZWq6yvbIFB7nA_i75q3lZuLlSIT8I8shiXeu1_W52RgAoA0DrSRNFd4BHmxcFVgsR3HOE92d52YPEG6cGggTW3Da2MSzPMB7WUZSO-TLEMxqYLYnTybCszX8wtByOTRILGCEcNayjnbMEt-ymTms0m0VJFN9EINLjFbfILbX7nq4PZshsaRpTOXpkGmLXPCaY-ldHzffeGrI8qQ%3D%3D&attredirects=0.

UNIWEBSIDAD. *1.1. Symfony en pocas palabras (Symfony 1.4, la guía definitiva)* [en línea]. Abril, 2018. [Consulta: 4 enero 2019]. Disponible en: <https://uniwebsidad.com/libros/symfony-1-4/capitulo-1/symfony-en-pocas-palabras?from=librosweb>.

VALLE, J. & GUTIERREZ, J. *Definición arquitectura cliente servidor* [en línea]. Julio, 2016. [Consulta: 8 enero 2019]. Disponible en: www.monografias.com.

VÍCTOR DÍAZ MARCO. *Conociendo WebSocket* [en línea]. 15 Noviembre, 2015. [Consulta: 6 junio 2019]. Disponible en: <https://v0ctor.me/websocket>.

VONICA, A. *Ventajas y desventajas de las aplicaciones web. | APLICACIONES WEB* [en línea]. 2 Octubre, 2016. [Consulta: 5 enero 2019]. Disponible en: <https://lasaplicacioneswebblog.wordpress.com/2016/10/02/ventajas-y-desventajas-de-las-aplicaciones-web/>.

YARIF JADISHA. *ISO 9126 | Estándares del Software* [en línea]. 24 Mayo, 2016. [Consulta: 23 abril 2019]. Disponible en: <https://estandarsw.wordpress.com/category/iso/iso-9126/>.

ZONE, J. *Tomcat, WebSockets, HTML5, jWebSockets, JSR-340, JSON y más - DZone Java* [en línea]. 23 Junio, 2017. [Consulta: 8 junio 2019]. Disponible en: <https://dzone.com/articles/tomcat-websockets-html5>.