



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

**“DESARROLLO DE UNA APLICACIÓN WEB CON EL  
FRAMEWORK BOOTSTRAP Y EL PRECOMPILADOR SASS  
PARA LA GESTIÓN DE PEDIDOS DE PRODUCTOS AGRÍCOLAS  
DE LA EMPRESA EL CHAGRA”**

**Trabajo de titulación**

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

**INGENIERO EN SISTEMAS INFORMÁTICOS**

**AUTOR: EDWIN ALFONSO HERNÁNDEZ BERRONES**

**DIRECTOR: ING. PAÚL XAVIER PAGUAY SOXO**

Riobamba – Ecuador

2020

**@2020, Edwin Alfonso Hernández Berrones**

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.

Yo, EDWIN ALFONSO HERNÁNDEZ BERRONES declaro que el presente trabajo de titulación es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 21 de noviembre del 2019

**Edwin Alfonso Hernández Berrones**

**060396063-4**

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**CARRERA DE INGENIERÍA EN SISTEMAS**

El Tribunal del trabajo de titulación certifica que: El trabajo de titulación: Tipo: Técnico, **“Desarrollo de una aplicación web con el Framework Bootstrap y el precompilador Sass para la gestión de pedidos de productos agrícolas de la Empresa El Chagra”**, realizado por el señor: **Edwin Alfonso Hernández Berrones**, ha sido minuciosamente revisado por los Miembros del Tribunal del trabajo de titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

**NOMBRE**

**FIRMA**

**FECHA**

Ing. Jorge Ariel Menéndez Verdecía  
**PRESIDENTE DEL TRIBUNAL**

\_\_\_\_\_

\_\_\_\_\_

Ing. Paúl Xavier Paguay Soxo  
**DIRECTOR DEL TRABAJO DE  
TITULACIÓN**

\_\_\_\_\_

\_\_\_\_\_

Ing. Mayra Alejandra Oñate Andino  
**MIEMBRO DEL TRIBUNAL**

\_\_\_\_\_

\_\_\_\_\_

## **DEDICATORIA**

Dedico este trabajo de titulación primero a Dios por brindarme una nueva oportunidad cada día de mi vida, a mis padres Alfonso Hernández Sislema y María Jesús Berrones por el apoyo incondicional y ser la fuente de inspiración, a mis hermanos María Belén y Brayan Kevin por sus palabras y consejos, a mis amigas Adriana y Elizabeth por demostrarme una amistad sincera a todos ustedes les dedico este esfuerzo en la culminación de mi carrera.

**Edwin.**

## **AGRADECIMIENTOS**

Gracias primeramente a Dios por ser el centro que guía mi vida, por su gracia que no la merezco y su inmensa fidelidad, demostrándome que siempre ha estado a mi lado y que tiene el control de todo.

También a la Escuela Superior Politécnica de Chimborazo, a la Escuela de Ingeniería en Sistemas y a todos sus profesores que me prepararon científicamente para ser un buen profesional, en especial al Ing. Paúl Paguay, por la guía durante el proceso de desarrollo de este trabajo con su conocimiento y experiencia, a la Ing. Alejandra Oñate por sus recomendaciones en la realización del mismo.

Con todo el corazón les agradezco.

**Edwin**

## TABLA DE CONTENIDO

ÍNDICE DE TABLAS .....	x
ÍNDICE DE FIGURAS .....	xi
ÍNDICE DE GRÁFICOS .....	xii
ÍNDICE DE ANEXOS .....	xiii
RESUMEN .....	xiv
ABSTRACT .....	xv
INTRODUCCIÓN.....	1

## CAPÍTULO I

1. MARCO TEÓRICO REFERENCIAL.....	6
1.1. Aplicación web.....	6
1.1.1. <i>Características</i> .....	6
1.1.2. <i>Ventajas y desventajas de las aplicaciones web</i> .....	7
1.2. Arquitectura de la aplicación web .....	7
1.2.1. <i>Comparativa entre las arquitecturas de software</i> .....	8
1.3. Arquitectura cliente-servidor .....	9
1.3.1. <i>Características</i> .....	9
1.3.2. <i>Ventajas y desventajas de la arquitectura cliente-servidor</i> .....	10
1.4. Patrón de Diseño.....	10
1.4.1. <i>Comparativa entre los patrones de diseño</i> .....	12
1.5. Patrón de Diseño Modelo Vista Controlador (MVC).....	12
1.5.1. <i>Características del patrón MVC</i> .....	13
1.5.2. <i>Ventajas y Desventajas del patrón de diseño Modelo Vista Controlador</i> .....	14
1.6. Framework para maquetado y scripting.....	14
1.6.1. <i>Comparativa entre frameworks para maquetado y scripting</i> .....	15
1.7. Framework Bootstrap .....	16

1.7.1.	<i>Características</i> .....	17
1.7.2.	<i>Ventajas y Desventajas</i> .....	17
1.8.	<b>Precompilador de las hojas de estilo Css</b> .....	17
1.8.1.	<i>Comparativa de los precompiladores más utilizados en css</i> .....	18
1.9.	<b>Precompilador Sass</b> .....	19
1.9.1.	<i>Características</i> .....	19
1.9.2.	<i>Ventajas y desventajas de Sass</i> .....	20
1.10.	<b>Lenguaje de programación</b> .....	20
1.10.1.	<i>Comparativa de los lenguajes de programación</i> .....	21
1.11.	<b>Lenguaje de programación PHP</b> .....	22
1.11.1.	<i>Características del lenguaje PHP</i> .....	22
1.11.2.	<i>Ventajas y desventajas del lenguaje de programación PHP</i> .....	23
1.12.	<b>Base de Datos</b> .....	23
1.12.1.	<i>Características</i> .....	24
1.12.2.	<i>Ventajas y desventajas de las bases de datos</i> .....	24
1.12.3.	<i>Comparativa de las bases de datos</i> .....	26
1.13.	<b>Sistema Gestor de base de datos MySql</b> .....	26
1.13.1.	<i>Características de SGBD MySql</i> .....	27
1.14.	<b>Metodología de desarrollo de software</b> .....	27
1.14.1.	<i>Comparativa de metodologías de software</i> .....	28
1.15.	<b>Metodología SCRUM</b> .....	29
1.15.1.	<i>Roles de la metodología SCRUM</i> .....	29
1.15.2.	<i>Proceso para el desarrollo del software</i> .....	30
1.15.3.	<i>Beneficios de la metodología SCRUM</i> .....	31
1.16.	<b>Estándar de evaluación del software</b> .....	32
1.16.1.	<i>Norma ISO 9126</i> .....	32
1.16.2.	<i>Factores de calidad</i> .....	33
1.16.3.	<i>Eficiencia de la Norma Iso 9126</i> .....	34

## CAPÍTULO II

<b>2. MARCO METODOLÓGICO</b> .....	<b>35</b>
<b>2.1. Diseño de la investigación</b> .....	<b>35</b>
2.1.1. <i>Tipo de investigación</i> .....	35
2.1.2. <i>Métodos de investigación</i> .....	35
2.1.3. <i>Técnicas de investigación</i> .....	36
<b>2.2. Determinación de los procesos que se manejan en la Empresa “El Chagra”</b> .....	<b>36</b>
2.2.1. <i>Diagrama de procesos no automatizado</i> .....	36
2.2.2. <i>Diagrama de procesos automatizado</i> .....	39
<b>2.3. Fase de planificación</b> .....	<b>42</b>
2.3.1. <i>Personas involucradas</i> .....	42
2.3.2. <i>Tipos y roles de usuario</i> .....	42
2.3.3. <i>Actividades realizadas en el proyecto</i> .....	42
2.3.4. <i>Product backlog</i> .....	43
2.3.5. <i>Sprint backlog</i> .....	46
2.3.6. <i>Reuniones SCRUM</i> .....	48
<b>2.4. Fase de desarrollo</b> .....	<b>48</b>
2.4.1. <i>Arquitectura del sistema</i> .....	48
2.4.2. <i>Estándar de codificación</i> .....	49
2.4.3. <i>Diseño de la base de datos</i> .....	49
2.4.4. <i>Diseño de la interfaz del sistema</i> .....	50
2.4.5. <i>Documentación</i> .....	52
2.4.6. <i>Manual de usuario</i> .....	53
<b>2.5. Fase de cierre</b> .....	<b>53</b>
<b>2.6. Gestión del proyecto</b> .....	<b>54</b>
2.6.1. <i>Burndown Chart</i> .....	54
<b>2.7. Método utilizado para la evaluación de la eficiencia</b> .....	<b>55</b>
2.7.1. <i>Población</i> .....	55
2.7.2. <i>Muestra</i> .....	55
2.7.3. <i>Obtención de datos</i> .....	56

### **CAPÍTULO III**

<b>3.</b>	<b>MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS DE RESULTADOS....</b>	<b>57</b>
<b>3.1.</b>	<b>Análisis de los procesos en la Actualidad de la Empresa .....</b>	<b>57</b>
<b>3.1.1.</b>	<i>Análisis de Normalidad .....</i>	<b>57</b>
<b>3.1.2.</b>	<i>Estadística Descriptiva .....</i>	<b>60</b>
<b>3.1.3.</b>	<i>Estadística Inferencial.....</i>	<b>61</b>
	<b>CONCLUSIONES.....</b>	<b>64</b>
	<b>RECOMENDACIONES.....</b>	<b>65</b>
	<b>BIBLIOGRAFÍA</b>	
	<b>ANEXOS</b>	

## ÍNDICE DE TABLAS

<b>Tabla 1-1:</b>	Ventajas y desventajas de las aplicaciones web .....	7
<b>Tabla 2-1:</b>	Tabla comparativa entre las arquitecturas de software .....	8
<b>Tabla 3-1:</b>	Ventajas y desventajas de la arquitectura cliente-servidor .....	10
<b>Tabla 4-1:</b>	Tabla comparativa entre patrones de diseño.....	12
<b>Tabla 5-1:</b>	Ventajas y Desventajas del patrón MVC.....	14
<b>Tabla 6-1:</b>	Comparativa entre framework de maquetado y scripting .....	16
<b>Tabla 7-1:</b>	Ventajas y Desventajas del framework Bootstrap .....	17
<b>Tabla 8-1:</b>	Comparativa de los precompiladores más utilizados en css .....	19
<b>Tabla 9-1:</b>	Ventajas y desventajas de Sass .....	20
<b>Tabla 10-1:</b>	Comparativa de los lenguajes de programación .....	21
<b>Tabla 11-1:</b>	Ventajas y desventajas de Php.....	23
<b>Tabla 12-1:</b>	Ventajas y desventajas de las bases de datos .....	24
<b>Tabla 13-1:</b>	Comparativa de las bases de datos.....	26
<b>Tabla 14-1:</b>	Comparativa de metodología de software .....	28
<b>Tabla 1-2:</b>	Personas involucradas en el desarrollo del proyecto .....	42
<b>Tabla 2-2:</b>	Tipos de usuarios y roles .....	42
<b>Tabla 3-2:</b>	Actividades realizadas en el proyecto.....	43
<b>Tabla 4-2:</b>	Método T-Shirt .....	44
<b>Tabla 5-2:</b>	Product Backlog.....	44
<b>Tabla 6-2:</b>	Sprint Backlog .....	46
<b>Tabla 7-2:</b>	Estándar de codificación.....	49
<b>Tabla 8-2:</b>	HU-01 Ingresar información de producto.....	52
<b>Tabla 9-2:</b>	Prueba de Aceptación, HU-01 Ingresar información de producto.....	52
<b>Tabla 10-2:</b>	Tarea de Ingeniería, HU-01 Ingresar información de producto.....	53
<b>Tabla 11-2:</b>	Actividades de la conclusión de la aplicación web.....	54
<b>Tabla 1-3:</b>	Resultados para la toma de decisión .....	59
<b>Tabla 2-3:</b>	Estadística descriptiva de los registros .....	59
<b>Tabla 3-3:</b>	Distribución Z con relación a los registros .....	60

## ÍNDICE DE FIGURAS

<b>Figura 1-1:</b> Partes de una aplicación web .....	6
<b>Figura 2-1:</b> Arquitectura cliente/servidor.....	9
<b>Figura 3-1:</b> Patrón MVC .....	13
<b>Figura 4-1:</b> Características de la ISO 9126 .....	33
<b>Figura 1-2:</b> Diagrama de procesos no automatizados de la gestión de pedidos.....	38
<b>Figura 2-2:</b> Diagrama de procesos Automatizados .....	41
<b>Figura 3-2:</b> Arquitectura de la aplicación web.....	48
<b>Figura 4-2:</b> Diagrama del modelo físico de la base de datos .....	50
<b>Figura 5-2:</b> Bosquejo de gestión de usuarios, productos, proveedores .....	51
<b>Figura 6-2:</b> Página de Inicio.....	51
<b>Figura 7-2:</b> Panel del administrador.....	52
<b>Figura 1-3:</b> Tiempos manuales de los registros.....	57
<b>Figura 2-3:</b> Tiempos automatizados de los registros.....	58

## ÍNDICE DE GRÁFICOS

<b>Gráfico 1-2:</b> Burndown Chart .....	54
<b>Gráfico 1-3:</b> Gráfica de Distribución Normal .....	61
<b>Gráfico 2-3:</b> Tiempo promedios de los registros.....	62

## **ÍNDICE DE ANEXOS**

**ANEXO A** Informe sobre reuniones equipo desarrollo-cliente

**ANEXO B** Estándar de programación

**ANEXO C** Estándar de interfaces

**ANEXO D** Manual Técnico

**ANEXO E** Manual de Usuario

**ANEXO F** Tiempos de Respuesta

**ANEXO G** Estadístico de Prueba

**ANEXO H** Estadística Descriptiva e Inferencial

**ANEXO I** Tiempos obtenidos de los requerimientos

## RESUMEN

El objetivo del presente trabajo de titulación fue el desarrollo de una aplicación web para la gestión de pedidos de productos agrícolas para la empresa El Chagra de la ciudad de Machachi. Para la recolección de requerimientos se utilizó la entrevista, la observación para el análisis de los procesos manuales de la empresa y los tiempos de respuesta, se utilizó la arquitectura cliente/servidor con el patrón de diseño Modelo Vista Controlador, mediante la utilización de herramientas como el lenguaje de programación php, el sistema gestor de base de datos MySQL, el framework de maquetación Bootstrap y el precompilador Sass. También se aplicó la metodología de desarrollo ágil Scrum, para la fase de planificación se obtuvieron nueve (9) sprints, veinte y ocho (28) historias de usuario y seis (6) historias técnicas, la fase de desarrollo tuvo una duración de setecientos veinte (720) horas desarrollando setenta y seis (76) tareas de ingeniería y ciento veinte (120) pruebas de aceptación siendo todas exitosas. Para la validación de la eficiencia se utilizó el estándar de ISO/IEC 9126, para medir el comportamiento de tiempos de respuesta de los procesos de registrar cliente, producto, proveedor y pedido; en la evaluación de la eficiencia de los procesos analizados se obtuvo una reducción de 57,81% con el nivel de significancia de 0,06. Para garantizar el correcto funcionamiento del sistema se recomienda a los usuarios finales utilizar el sistema conforme se establece en el manual de usuario.

**PALABRAS CLAVE:** <INGENIERIA DE SOFTWARE>, <DESARROLLO DE SOFTWARE>, <GESTIÓN DE PEDIDOS>, <FRAMEWORK BOOTSTRAP>, <PRECOMPILADOR SASS>, <METODOLOGÍA DE DESARROLLO ÁGIL (SCRUM)>, <SISTEMA GESTOR DE BASE DE DATOS (MYSQL)>, <MODELO VISTA CONTROLADOR>, <NORMA ISO/IEC 9126>.

## ABSTRACT

The objective of this research work was the development of a web Application for order management of agricultural products for the Company El Chagra in the city of Machachi. For the collection of requirements, the interview was used, the observation for the analysis of the manual processes of the company and the response times, the client / server architecture was used with the Design Controlled Model design pattern, using tools such as the php programming language, the MySql database management system, the Bootstrap layout framework and the Sass compiler. Agile development methodology Scrum was also applied, for the planning phase nine (9) sprints were obtained, twenty-eight (28) user stories and six (6) technical stories and the development phase lasted seven hundred twenty (720) hours developing seventy-six (76) engineering tasks and one hundred twenty (120) acceptance tests being all successful for the validation of efficiency, the standard of ISO/IEC 9216 was used to measure the response time behavior of the processes of provide customer, product, supplier and order; in the evaluation of the efficiency of the processes analysed, a reduction of 57.81% was obtained with the level of significance of 0,06. To ensure the proper functioning of the system, end users are recommended to use the system as set out in the user manual.

**Keywords:** <SOFTWARE ENGINEERING>, <SOFTWARE DEVELOPMENT>, <ORDER MANAGEMENT>, <FRAMEWORK BOOTSTRAP>, <SASS PRE-COMPILER>, <AGILE DEVELOPMENT METHODOLOGY (SCRUM)>, <DATA BASE MANAGEMENT SYSTEM (MYSQL)>, <CONTROLLER VIEW MODEL>, <EFFICIENCY>

## INTRODUCCIÓN

La humanidad en nuestros días ha tenido avances tanto de manera social como en el ámbito de la tecnología que facilita el convivir entre las personas y regula sus actividades mejorando la comunicación y comercialización, donde las empresas muestran sus productos a su distinguida clientela con el fin de darse a conocer a nivel provincial o nacional.

La prioridad de las empresas es convertirse en líderes en cuanto a servicios prestados, desarrollando una mejor actividad comercial con el acogimiento de clientes que puedan conocer los productos ofrecidos de forma diferente mediante la utilización de la tecnología.

Una aplicación web facilita la manipulación de información de las empresas mediante la automatización de sus procesos, mejorando de manera notable los tiempos de atención a los clientes.

La Empresa El Chagra se ubica en la ciudad de Machachi y tiene como actividad la comercialización de productos agrícolas, realizando sus actividades de manera manual tomando mucho tiempo en la atención al cliente lo que se podría mejorar mediante el desarrollo de una aplicación web para automatizar los procesos de pedidos con la finalidad de prestar mejores servicios y reducir el tiempo de atención.

Para lo cual se ha tomado en cuenta los módulos necesarios que formen parte de esta aplicación web los cuales se mencionan a continuación.

El **módulo usuarios** tiene la finalidad de manejar los usuarios que puedan ingresar al sistema y su gestión, en el **módulo clientes** se podrá gestionar la información del cliente registrado en el sistema, en el **módulo de productos** consta toda la información para llevar el control de los productos que posee la empresa, en el **módulo de reportes** se tendrá la opción de poder revisar información relevante sobre la empresa sobre los productos, proveedores y pedidos realizados, en el **módulo inicio** de la empresa se manejará datos relevantes como su misión, visión y marcas de los productos agrícolas, en el **módulo de proveedores** se gestiona los datos de los proveedores de los productos agrícolas de la empresa, en el **módulo pedidos** se gestionara la información referente al pedido que el cliente ha realizado.

## **Antecedentes**

Los framework web son un conjunto de herramientas informáticas de librerías y estilos dispuestas en una estructura que tiene una específica base, para el desarrollo de aplicaciones web tanto sencillas como escalables (Navarrete et al, 2017, p. 2).

Hace algún tiempo el uso de framework eran utilizados en programación pura, en la actualidad es muy utilizado los framework para maquetado y scripting, específicamente en áreas correspondientes a un desarrollo front-end. Existen herramientas como Bootstrap que brindan un conjunto de elementos como la implementando de html, Css y javascript para el desarrollo, permitiendo la visualización de sitios web atractivos.

Las nuevas tendencias tecnológicas brindan un soporte a entidades dedicadas al comercio mejorando notablemente su producción, comercialización de los bienes que estas ofrecen, se nota una gran variedad de ventajas las cuales pueden ser:

- Reducción de costos y tiempo.
- Acceso a los datos.
- Integridad de los datos almacenados.
- Rapidez en procesos realizados.

La Empresa “El Chagra” tiene como prioridad gestionar la respectiva información de cada uno de los productos que ofrece, de manera que toda esta información no se encuentra automatizada lo cual provoca demora en la gestión de pedidos de productos agrícolas por parte del usuario.

Por lo tanto, en el presente proyecto se plantea el desarrollo de una aplicación web para la gestión de la información del proceso de pedidos de productos agrícolas de la Empresa “EL Chagra”.

## **Formulación del problema**

Actualmente la Empresa “EL CHAGRA” de la provincia de Pichincha ubicado en las calles Antonio Benites y 10 de agosto, presenta un proceso no automatizado de la información, lo cual provoca demora en la gestión de pedidos de productos agrícolas por parte de los clientes.

Teniendo esta información de la Empresa “EL CHAGRA”, es necesario desarrollar una aplicación web para la gestión de productos agrícolas.

¿La aplicación web para la Empresa “EL CHAGRA” mejorará la eficiencia en la gestión de los procesos de manejo de los productos agrícolas?

### **Sistematización del problema**

¿Cómo se lleva a cabo en la actualidad la gestión de la información del proceso de pedidos de los productos agrícolas en la Empresa “El Chagra”?

¿Cuál es el patrón de diseño más adecuado para la aplicación a desarrollar?

¿Qué herramientas se podrían aplicar para el desarrollo de la aplicación web?

¿Cómo mejorará la eficiencia en la gestión de la información del proceso de pedidos de los productos agrícolas en la Empresa “El Chagra”?

### **Justificación del trabajo de titulación**

#### ***Justificación Teórica***

Se denomina aplicación web al software que los usuarios pueden utilizar accediendo a un servidor web a través de internet mediante un navegador. Las aplicaciones web son utilizadas debido a la navegación web como un cliente ligero, también puede ser independiente del sistema operativo que esté utilizando, así la facilidad para actualizar y mantener aplicaciones sin distribuir e instalarlas en los ordenadores de sus usuarios.

Una aplicación web debe contener elementos que permitan una comunicación activa entre el usuario y la información. Por ejemplo, rellenar y enviar formularios, responder a determinadas acciones ofrecido a continuación datos y la información previamente almacenados en una base de datos.

El desarrollo de aplicaciones web es una de las competencias de un ingeniero en sistemas, por ello se pretende realizar una aplicación web ya que se encuentra dentro del campo de desarrollo de software.

Con la finalidad de desarrollar la aplicación web para la gestión de los procesos de manejo de productos para la Empresa “El Chagra” mediante el empleo de lenguaje de programación PHP

el cual es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor, así también el motor de base de datos MySQL siendo un servidor de base de datos que incluye características de la orientación a objetos como puede ser la herencia, tipos de datos, funciones, restricciones, reglas y para el diseño de las interfaces se usará el framework Bootstrap que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web. También se usará SAAS siendo este un pre-compilador para la incrustación de las clases CSS de Bootstrap directamente en HTML teniendo la ventaja de ser más rápido.

### ***Justificación Aplicativa***

Es necesario el desarrollo de una aplicación web ya que permitirá la gestión directa de los datos de los productos, proveedores, clientes y pedidos, lo cual ayuda a obtener información de forma rápida al personal encargado de atender al agricultor. Así también brindará un entorno gráfico usable, agradable y escalable.

Las funciones básicas de este sistema radican en el control de la información del proceso de los productos, proveedores, clientes y pedidos. Dentro de este marco, la meta es simplificar las tareas realizadas en la Empresa “El Chagra”.

Los módulos a implementar son los siguientes:

- Módulo de Gestión del Usuario
- Módulo de Gestión Cliente
- Módulo de Gestión Producto
- Módulo de Reportes
- Módulo de Gestión Proveedores
- Módulo de Gestión Pedido

El desarrollo de dicha aplicación web se encuentra acorde a las líneas de investigación de la EIS. Proceso de desarrollo de software. Así como a las líneas de investigación de la ESPOCH referente a las Tecnologías de la Información, comunicación, procesos industriales y biotecnología acorde al programa para el desarrollo de aplicaciones de software para procesos de gestión y administración pública y privada.

Y bajo los parámetros del Programa Nacional Toda una vida.

## **Objetivos**

### **Objetivo General**

Desarrollar una aplicación web con el framework Bootstrap y el precompilador Sass para la gestión de pedidos de productos agrícolas de la Empresa “El Chagra”.

### **Objetivos Específicos**

- Analizar la situación actual de los procesos para la gestión de productos agrícolas de la Empresa “El Chagra”.
- Diseñar la aplicación web utilizando el patrón MVC y la metodología ágil Scrum.
- Implementar en la aplicación web el framework Bootstrap con el pre-compilador Sass.
- Evaluar la eficiencia en la aplicación web, utilizando la norma ISO 9126.

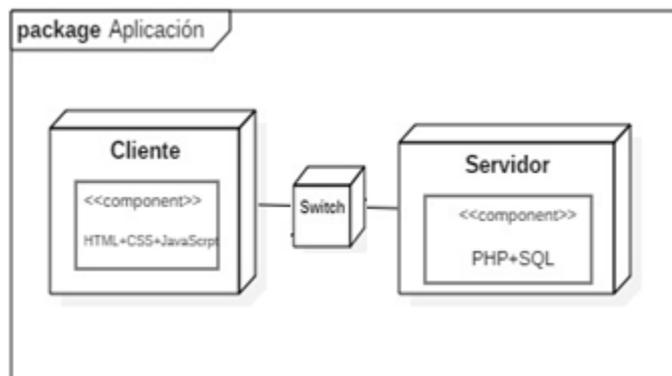
# CAPÍTULO I

## 1. MARCO TEÓRICO REFERENCIAL

### 1.1. Aplicación web

Una aplicación web es una herramienta informática que se puede acceder desde cualquier navegador, ya sea desde el internet o también desde una red local; teniendo la facilidad de acceder a cualquier funcionalidad que se requiera (Blasco, 2013, p.10).

Teniendo un gran auge debido a lo práctico del navegador web y a la previsión de la tasa de crecimiento anual del 33% hasta el año 2021 lo que permite agilidad, diseño atrayente y funciona en todo dispositivo y plataforma de distribución (Mora, 2012, p.2).



**Figura 1-1:** Partes de una aplicación web  
Realizado por: Edwin Hernández, 2019

#### 1.1.1. Características

La aplicación web posee una serie de características que se los puede diferenciar de otras aplicaciones de software, y son:

- Desde el punto de vista del usuario: en cuanto a la usabilidad no depende que sea un usuario experto o con habilidades limitadas; lo que facilita a los usuarios el uso de una aplicación para el acceso de la información (Pastor, 2013, p.5).

- Desde el punto de vista de la plataforma: se utiliza la red local o la conexión a internet y se puede acceder desde distintos dispositivos (Pastor, 2013, p. 5).
- Desde el punto de vista de la información: existe una disponibilidad de fuentes heterogéneas de información, las cuales pertenecen a diferentes dominios que colaboran con la aplicación (Pastor, 2013, p. 6).

### 1.1.2. Ventajas y desventajas de las aplicaciones web

En la **Tabla 1-1**, se describe las ventajas y desventajas de las aplicaciones web que se muestran a continuación:

**Tabla 1-1:** Ventajas y desventajas de las aplicaciones web

Ventajas	Desventajas
<ul style="list-style-type: none"> <li>• Implementado en diferentes sistemas operativos.</li> <li>• Resulta fácil la curva de aprendizaje, es entendible su código.</li> <li>• Puede ser utilizada para diferentes tareas por diferentes usuarios.</li> </ul>	<ul style="list-style-type: none"> <li>• Necesitan obligatoriamente de conexión a internet.</li> <li>• El acceso que se tiene a las características de hardware del dispositivo es limitado.</li> <li>• Toma un mayor esfuerzo en realizar publicidad y visibilidad.</li> </ul>

Fuente: (Glera, 2013, pp. 25-26)

Realizado por: Edwin Hernández, 2019

Mediante la información analizada, el presente trabajo de titulación se decidió implementar una aplicación web dado que en la actualidad el uso del internet es necesario para la comunicación del dueño de la empresa “El Chagra” y sus clientes; mediante esto se podrá tener un mejor control desde cualquier lugar de los procesos realizados en la empresa **Tabla 1-1**, y ayudará en la administración de los clientes, productos, proveedores y pedidos.

## 1.2. Arquitectura de la aplicación web

La arquitectura en cuanto a la relación con aplicaciones web está formada por un conjunto de varios tipos de factores, arreglos y normas que interactúan entre sí para brindar un servicio informático al usuario (Gutierrez, 2005, p.3).

En la actualidad existen varias arquitecturas de software, entre las más usadas para el desarrollo de aplicaciones web se encuentra monolítica, cliente-servidor, n capas (Marini, 2012, p. 10).

### Monolítica

Es aquella arquitectura completamente independiente, puede interactuar con otros servicios o datos en el transcurso de sus operaciones; pero el núcleo de su comportamiento se ejecuta dentro de su propio proceso y toda la aplicación se implementa como una única unidad. (Peña, 2014, <https://prezi.com/2uraxobilh0t/tipos-de-arquitecturas-de-software/>).

### **Ciente-servidor**

Es una arquitectura de comunicación de computadores definidos como cliente y servidor, el computador cliente solicita el servicio al computador servidor por medio de un mensaje definido; la diferencia es que el cliente es el que inicia el contacto mientras que el servidor es el que responde a la petición de conexión (Gutierrez, 2005, p.4).

### **N capas**

En la programación por capas el objetivo principal es la separación de la lógica de negocios de la lógica de diseño; separando de esta manera los datos almacenados en el gestor de bases de datos utilizado (Peña, 2014, <https://prezi.com/2uraxobilh0t/tipos-de-arquitecturas-de-software/>).

#### ***1.2.1. Comparativa entre las arquitecturas de software***

En la **Tabla 2-1**, se muestra la comparativa entre las arquitecturas de software monolítica, cliente-servidor, n capas siendo estas las más utilizadas para el desarrollo de aplicaciones web; tomando en cuenta los parámetros como: escalabilidad maneja el crecimiento continuo de trabajo de manera fluida, facilidad de mantenimiento la aplicación es susceptible de ser corregida, flexibilidad medida que la aplicación puede realizarse cambios, versatilidad adaptarse con rapidez a distintas funciones y costos los gastos que representa utilizar una cierta arquitectura.

**Tabla 2-1:** Tabla comparativa entre las arquitecturas de software

<b>PARÁMETROS</b>	<b>MONOLÍTICA</b>	<b>CLIENTE-SERVIDOR</b>	<b>N CAPAS</b>
Escalabilidad	NO	SI	SI
Facilidad de mantenimiento	NO	SI	SI
Flexibilidad	NO	SI	SI
Versatilidad	NO	SI	SI

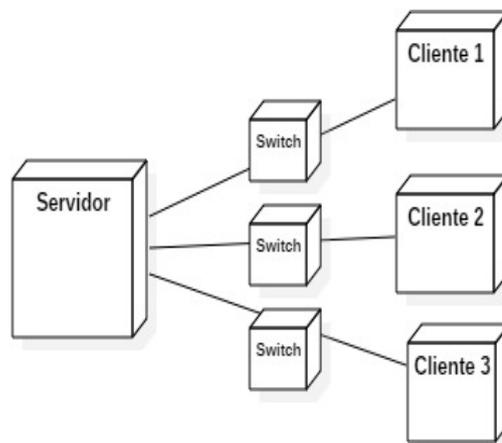
Costos	Altos	Medio	Alto
--------	-------	-------	------

**Fuente:** (Ruiz, 2017, [https://sandramarramirez.fandom.com/es/wiki/Arquitectura\\_Monolítica\\_y\\_Arquitectura\\_Cliente\\_-\\_Servidor](https://sandramarramirez.fandom.com/es/wiki/Arquitectura_Monolítica_y_Arquitectura_Cliente_-_Servidor))  
**Realizado por:** Edwin Hernández, 2019

Una vez realizado el análisis de las arquitecturas de software más utilizadas para el desarrollo de una aplicación web y el estudio comparativo entre ellas realizado en la **Tabla 2-1**, se ha escogido para el desarrollo del presente trabajo de titulación la arquitectura cliente-servidor por los costos en el momento de su utilización y las ventajas que presenta con respecto a las demás arquitecturas de software.

### 1.3. Arquitectura cliente-servidor

Es un arquitectura en la que las diferentes tareas son asignadas entre los proveedores y los demandantes o también conocidos como servidores y clientes, que realizan peticiones a uno o más servidores los cuales se encuentran en ejecución para atender la demanda; permitiendo diversificar el trabajo de cada aplicación de esta form los clientes no se sobrecargan de trabajo (Marini, 2012, p.1).



**Figura 2-1:** Arquitectura cliente/servidor  
**Realizado por:** Edwin Hernández, 2019

#### 1.3.1. Características

La arquitectura cliente-servidor posee varias características que se puede diferencias de las demás arquitecturas, las cuales son:

- El servidor atiende a las peticiones de los clientes con una interfaz única y bien definida (Orfali, 2002, p. 17).
- El cliente no está sujeto a la ubicación de cada servidor, ni el equipo de ejecución; es independiente del sistema operativo (Orfali, 2002, p. 17).
- Las peticiones de varios clientes pueden ser atendidos por un servidor al mismo tiempo, regulando sus accesos a los recursos (Orfali, 2002, p. 17).
- Siempre los clientes inician la comunicación pidiendo el servicio, esperando pasivamente las peticiones de cada cliente (Orfali, 2002, p.18).
- El mensaje es el canal de comunicación de las peticiones de servicio de cada cliente al servidor (Orfali, 2002, p. 19).
- Los datos del servidor están centralizados, mejorando costos, mantenimiento y el control de la integridad (Orfali, 2002, p. 19).

### 1.3.2. Ventajas y desventajas de la arquitectura cliente-servidor

Las ventajas y desventajas de la arquitectura cliente-servidor se muestran en la **Tabla 3-1**, a continuación:

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

Ventajas	Desventajas
<ul style="list-style-type: none"> <li>• Facilidad en la integración de los diferentes sistemas, permitiendo compartir información.</li> <li>• Permite al usuario tener mayor interacción con la aplicación.</li> <li>• Genera un orden de trabajo, donde cada cliente puede acceder al mismo servidor sin generar conflictos.</li> <li>• Facilita la integración de nuevas funcionalidades, favoreciendo la estabilidad de las aplicaciones web.</li> </ul>	<ul style="list-style-type: none"> <li>• Necesaria habilidad para reparar un servidor, sin perder la información y el correcto funcionamiento.</li> <li>• En la seguridad se requiere pasar por procesos de validación, mediante protocolos.</li> </ul>

Fuente: (Schiaffarino, 2019, <https://blog.infranetworking.com/modelo-cliente-servidor/>)

Realizado por: Edwin Hernández, 2019

## 1.4. Patrón de Diseño

El patrón de diseño es una solución a un problema repetible en el diseño de software, la cual no es un diseño terminado que puede ser traducido a código sino una descripción de cómo resolverlo, que puede ser utilizado en varias situaciones; reflejando de esta manera el diseño y

modificación que los desarrolladores deben hacer para lograr mayor reutilización y flexibilidad en el software (Campo, 2009, p.103).

Los patrones explican problemas de diseño para luego discutir una solución, siendo estos incorporados al conocimiento y experiencia en la demandante industria del software; teniendo una gran relevancia en la práctica más que en la teoría (Campo, 2009, p. 103).

Identificando las clases e instancias participantes, diferenciando los roles y colaboraciones bajo las responsabilidades, cada patrón se centra en un problema concreto describiendo cuando aplicarlo con las debidas restricciones de diseño (Campo, 2009, p. 104).

Existen varios patrones de diseño, entre los más usados para el desarrollo de aplicaciones web son Modelo-Vista-Controlador (MVC), pizarra y filtro de tubería (Mestras, 2004, p.11).

### **Modelo-Vista-Controlador**

Este patrón se diferencia de los demás por la separación que hace en cuanto a sus componentes, teniendo bien definida la capa de presentación la cual interactúa con la capa de negocios y la capa de datos para el acceso a la información almacenada en un sistema gestor de base de datos (Arcos et al, 2018, p. 244).

### **Pizarra**

Este patrón es usado para problemas que no se conocen las estrategias de solución determinista, consta de varios elementos funcionales denominados agentes y un instrumento de control denominado pizarra; los agentes están especializados en una tarea específica y cooperan para alcanzar una meta común realizando sus tareas y escribir sus propias conclusiones en la misma pizarra (Huaman, 2018, <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>).

### **Filtro de tubería**

Este patrón es usado para sistemas que producen y procesan datos, los procesamientos se incluyen en el filtro, que conectan componentes a través de los conectores y las comunicaciones se ejecutan como un flujo; los datos se procesan a través de tuberías, siendo utilizadas para la sincronización de almacenamiento (Huaman, 2018, <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>).

#### 1.4.1. Comparativa entre los patrones de diseño

En la **Tabla 4-1**, se muestra la comparativa entre los patrones Modelo-Vista-Controlador, pizarra y filtro de tubería siendo estos los más utilizados para el desarrollo de aplicaciones web; tomando en cuenta parámetros como: escalabilidad la cual se refiere a la propiedad de aumentar el tamaño de un sistema, control manipula los recursos, solución mejora la eficiencia operativa, mantenibilidad representa la cantidad de esfuerzo requerida para mantener su funcionamiento normal, reusabilidad hace referencia a poder volver a usar parte de la aplicación web, seguridad en torno a la protección de la aplicación web y explicaciones que sea claro, entendible para su uso.

**Tabla 4-1:** Tabla comparativa entre patrones de diseño

PARÁMETROS	MVC	PIZARRA	FILTRO DE TUBERÍA
Escalabilidad	SI	NO	SI
Control	Los recursos del servidor	Depende del estado del pizarrón	Variables de transformación
Solución	A través del análisis de la situación real	No existe garantía	Mediante un flujo a través de conectores
Mantenibilidad	Fácil	Difícil	Difícil
Reusabilidad	SI	NO	SI
Seguridad	SI	NO	SI
Explicaciones	SI	NO	NO

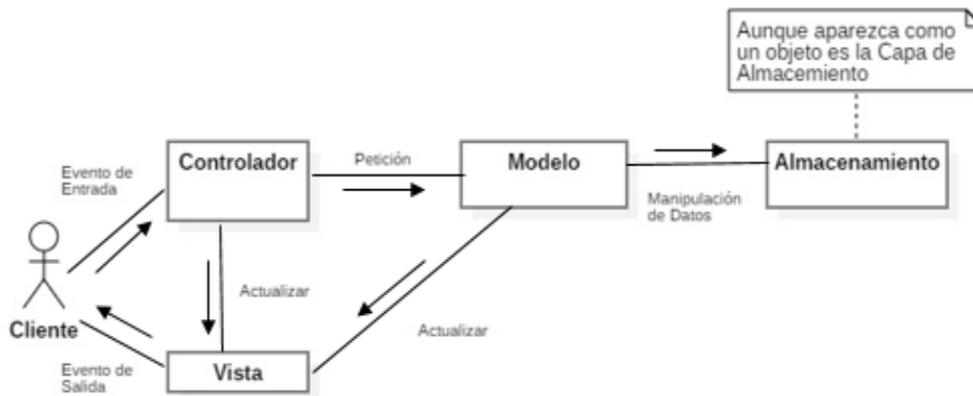
**Fuente:** (Fontela, 2010, p.11); (Rivera, 2010, p. 9); (Cristiá, 2006, p. 18)

**Realizado por:** Edwin Hernández, 2019

Después del análisis realizado con los patrones de diseño, se ha decidido implementar el patrón de diseño MVC debido a que permite estructurar y separar la interfaz con el controlador y el modelo; facilitando la mantenibilidad de la aplicación al momento de modificar las interfaces existentes de la aplicación web.

#### 1.5. Patrón de Diseño Modelo Vista Controlador (MVC)

En forma general Modelo-Vista-Controlador es un patrón de diseño de software para programación que propone separar el código por sus diferentes responsabilidades; descomponer la aplicación en capas permitiendo tener una separación entre la lógica de negocios de la aplicación, la representación y la persistencia (Mestras, 2008, pp. 4-5).



**Figura 3-1:** Patrón MVC  
Realizado por: Edwin Hernández, 2019

El patrón MVC identifica tres capas que son importantes para cualquier aplicación las cuales son:

- **Modelo:** es el módulo el cual tiene la lógica de negocios, por el cual se accede a los datos de la aplicación como intermedio con la base de datos y la lógica para interactuar entre ellos (Mestras, 2008, p. 6).
- **Vista:** módulo encargado de maneja la interacción con el usuario las interfaces de la aplicación web y la representación del modelo (Bascón, 2004, p. 495).
- **Controlador:** módulo que se encarga de ser el intermediario entre el modelo y la vista frente a las peticiones generadas por el cliente; el controlador selecciona el modelo solicitado por el usuario y la vista adecuada para representarlo (Mestras, 2008, p.7).

### 1.5.1. Características del patrón MVC

El patrón MVC posee varias características que se puede diferenciar de los demás patrones de diseño, los cuales son:

- **Transparencia de localización**

El funcionamiento del servidor web puede estar en el ordenador del cliente o en uno diferente, teniendo en cuenta que debe estar conectado en una misma red. Mediante esto el cliente puede no mostrar el servidor, pero direccionando las peticiones a donde se encuentra el servidor para brindar las peticiones solicitadas (Segovia, 2005, p.8).

- **Transparencia**

No se toma en cuenta el sistema operativo en donde se encuentre un sistema informático, los datos pueden ser migrados muy fácilmente entre servidores conectados permitiendo un trabajo continuo y asegurando la integridad de los datos (Segovia, 2005, p.9).

- **Escalabilidad**

Mejora el rendimiento de una aplicación web añadiendo servidores a los ya existentes a esto se le conoce como escalamiento horizontal, mientras que el escalamiento vertical es la migración de los datos a servidores más potentes (Garrido, 2004, p.5).

### 1.5.2. *Ventajas y Desventajas del patrón de diseño Modelo Vista Controlador*

Las ventajas y desventajas del patrón de diseño MVC se encuentran en la **Tabla 5-1**, entre las cuales tenemos las siguientes:

**Tabla 5-1:** Ventajas y Desventajas del patrón MVC

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none"> <li>• Desarrollo en menos tiempo y reutilización de las partes de forma independiente.</li> <li>• En cuanto a la migración cubriendo las altas demandas y permitir agregar nuevos componentes o reemplazar otros.</li> <li>• Facilita el manejo de los errores.</li> </ul>	<ul style="list-style-type: none"> <li>• En cuanto a la cantidad de archivos en mantener y desarrollar se incrementa.</li> <li>• La separación entre capas resulta en dificultad en el desarrollo del sistema.</li> </ul>

**Fuente:** (Gómez, 2015, <http://rodrigogr.com/blog/modelo-vista-controlador/> ; Bermeo, 2014, p.39 )  
**Realizado por:** Edwin Hernández, 2019

### 1.6. Framework para maquetado y scripting

Un framework es un conjunto de clases que cooperan para construir un diseño reutilizable para un específico tipo de software, proporciona la arquitectura partiendo del diseño en clases

abstractas con sus responsabilidades y colaboraciones; el desarrollador realiza una aplicación utilizando componentes y subclases a partir de las clases ya definidas por el framework (María et al, 2008, p. 38).

Hace algunos años el framework solo se utilizaba para programación pura, en la actualidad es muy aplicado los framework de scripting y maquetado; especialmente en las áreas correspondientes a un desarrollo front-end, implementando html, css y javascript para el desarrollo de sitios web atractivos (Tovar, 2013, <https://es.slideshare.net/GeraldynDeSousa/framework-30197256>).

Existen varios frameworks para maquetado y scripting, los más utilizados son Bootstrap, Foundation y Materialize (Alvarez, 2018, <https://desarrolloweb.com/articulos/materialize-framework-css.html>).

### **Bootstrap**

El framework Bootstrap es una biblioteca multiplataforma o conjunto de herramientas de código abierto utilizado en el diseño de aplicaciones web, mediante el uso de plantillas de diseño con botones, cuadros, menú de navegación y otros elementos de diseño basados en html y css (Valbuena, 2014, p. 23).

### **Foundation**

Es un framework front-end consta de herramientas para la toma de respuestas, construido con html, css y jquery utilizando tecnologías y prácticas modernas, cuenta con clases de marcador de posición permitiendo a los usuarios tener tiempo de carga muy cortos como sea posible (Valbuena, 2014, p. 26).

### **Materialize**

Es un framework que permite implementar de una manera muy sencilla las guías de diseño de material design, ofreciendo una base de css sobre la cual implementar cualquier proyecto; ya sea en un sitio web o aplicación basada en estándares web, teniendo una cantidad de interfaces ya diseñadas que pueden incorporarse a un proyecto (Alvarez, 2018, <https://desarrolloweb.com/articulos/materialize-framework-css.html>).

#### ***1.6.1. Comparativa entre frameworks para maquetado y scripting***

Entre los framework para maquetado y scripting más utilizados para el desarrollo de las interfaces de usuario están Bootstrap, Foundation y Materialize siendo los más utilizados para el

desarrollo de interfaces de usuario; que se encuentran en la **Tabla 6-1**, teniendo parámetro como son: sistema base de la interfaz, compatible con los navegadores, preprocesador es un programa separado que es invocado por el compilador, web aplicado para internet, descarga, comunidad sitio oficial de los frameworks, aprendizaje y open source código abierto.

**Tabla 6-1:** Comparativa entre framework de maquetado y scripting

PARÁMETROS	BOOTSTRAP	FUNDATION	MATERIALIZE
Sistema	12 columnas	1-16 columnas	12 columnas
Compatible con los navegadores	Chrome, Firefox, Safira, Opera	Chrome, Firefox, IE +, Safira	Chrome, Firefox, Opera
Preprocesador	Sass	Sass	Less
Web	SI	SI	SI
Descarga	Fácil	Complicada	Complicada
Comunidad	Oficial	No cuenta	No cuenta
Aprendizaje	Baja	Alta	Media
Open Source	SI	SI	SI

**Fuente:** (Valbuena, 2014, pp. 23-27; Ocaña, 2014, <https://www.azulweb.net/materialize-css-framework-responsive-google/>)

**Realizado por:** Edwin Hernández, 2019

Por la información recabada para este trabajo de titulación se ha decidido utiliza el framework Bootstrap por la rapidez de crear interfaces de usuario en aplicaciones web, permitiendo la personalización de manera sencilla; por las ventajas que ofrece de acuerdo a la comparativa de la **Tabla 6-1**, y principalmente porque utiliza el precompilador Sass siendo este el propuesto para el desarrollo de la aplicación web para la gestión de pedidos en la Empresa “El Chagra”.

### 1.7. Framework Bootstrap

Se conoce a Boostrap como el framework que ayuda a la creación de las interfaces de usuario mediante la utilización de componentes como CSS y JavaScript, mediante esto se puede visualizar en cualquier dispositivo porque se ajusta a las proporciones y resolución de la pantalla en la cual se observa a esto se le conoce como responsive Design (Acedo 2015, <http://programacion.jias.es/2015/05/web-¿que-es-el-framework-bootstrap-ventajas-desventajas/>).

Mejorando notablemente las proporciones de los CSS de las interfaces de usuario, ayudando al desarrollo web de las aplicaciones; teniendo una interfaz más amigable con el usuario y el incremento en la interacción con la misma desde cualquier lugar (Solis 2017, <https://www.arweb.com/chucherias/¿que-es-bootstrap-y-como-funciona-en-el-diseno-web/>).

### 1.7.1. Características

El framework Bootstrap posee varias características que se puede diferenciar de los demás frameworks de acuerdo a (Rodríguez, 2012, <https://www.genbeta.com/desarrollo/bootstrap>):

- Facilita la adaptación de la interface a diferentes navegadores.
- Se integra con las principales librerías de javascript.
- Ofrece un diseño agradable utilizando Sass con los estándares de css.
- Es muy ligero y adaptable a todo tipo de proyecto desarrollado.
- Dispone de distintos layout definidos con estructura fija a 940 píxeles de diferentes columnas o diseños fluidos.

### 1.7.2. Ventajas y Desventajas

Las ventajas y desventajas del framework Bootstrap se describen en la **Tabla 7-1**, la cual se muestra a continuación:

**Tabla 7-1:** Ventajas y Desventajas del framework Bootstrap

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none"> <li>• <b>Soporte:</b> Hay una gran comunidad de desarrolladores con implementación externa como WordPress.</li> <li>• <b>Rapidez y Comodidad:</b> Sencilla herramienta de construcción para sitios web e interfaces.</li> <li>• <b>Componentes:</b> Mucha variedad de temas y plantillas a utilizar.</li> </ul>	<ul style="list-style-type: none"> <li>• <b>Aprendizaje:</b> Comprensión y familiarizar con su estructura y nomenclatura.</li> <li>• <b>Mantenimiento:</b> Es complicado, cambiar de versión.</li> <li>• <b>Ampliar componentes:</b> Tener cuidado con el diseño.</li> </ul>

**Fuente:** (Acedo, 2015, <http://programacion.jias.es/2015/05/web-¿que-es-el-framework-bootstrap-ventajas-desventajas/>)

**Realizado por:** Edwin Hernández, 2019

## 1.8. Precompilador de las hojas de estilo Css

Un precompilador de Css es una herramienta que permite escribir pseudo código Css que luego será convertido a Css real. El cual se conforma de variables, condiciones, bucles o funciones. El objetivo es tener un código más sencillo de mantener y editar (Lozano, 2014, <https://abalozz.es/que-es-un-preprocesador-de-css/>).

Mediante la utilización de variables, las cuales almacenan valores y reutilizarlos en cualquier parte del código. También las funciones que se utiliza para evitar código duplicado, y los prefijos propietarios de cada navegador (Pérez, 2009, p. 57).

Actualmente existe varios precompiladores de hojas de estilo css entre los más utilizados están Sass, Less y PostCSS (Lozano, 2014, <https://abalozz.es/que-es-un-preprocesador-de-css/>).

### **Sass**

El preprocesador Sass agrega características elegantes y potentes a la interfaz de usuario, con la utilización de variables, Mixins, hojas de estilo; manteniendo el mismo trabajo y la compatibilidad con CSS (Ramírez, 2016, p. 1).

### **Less**

Es un preprocesador que facilita el tener un css menos complicado, añadiendo nuevas características muy útiles como las variables, mixins, operaciones y funciones como una alternativa para Sass; añadiendo comportamiento dinámico a css pudiendo ejecutarse tanto del lado del servidor como del cliente (Mendoza, 2015, <https://blog.michelletorres.mx/ventajas-del-uso-de-less/>).

### **PostCSS**

Herramienta que permite la utilización de plugins de JavaScript para los estilos, no alterando de ninguna manera los archivos css (Mazza, 2015, <https://octuweb.com/introduccion-a-postcss/>). Transforma las hojas css en una serie de datos que javascript pueda manipular, no cambiando el código solo hace que los plugins pueda cambiarlo no teniendo límite de manipulación (Ortega 2016, <https://openwebinars.net/blog/que-es-postcss-introduccion/>).

#### ***1.8.1. Comparativa de los precompiladores más utilizados en css***

En la **Tabla 8-1**, se muestra la comparativa entre los precompiladores más utilizados en css como son Sass, Less y PostCSS para el desarrollo de interfaces; con los parámetros que son: compatibilidad con las hojas de estilo css, configuración es que cumpla una función específica,

comunidad el sitio oficial, maduro es el tiempo de desarrollo, estable su nivel de fallos disminuye, potente trabaja bien en ciertas condiciones e implementación una de las etapas de desarrollo.

**Tabla 8-1:** Comparativa de los precompiladores más utilizados en css

PARÁMETROS	SASS	LESS	POSTCSS
Compatibilidad con css	Con todas las versiones	Con todas las versiones	-----
Configuración	Fácil	Medio	Difícil
Comunidad	Oficial	No Oficial	No oficial
Maduro	SI	SI	SI
Estable	SI	NO	NO
Potente	SI	SI	NO
Implementación	Fácil	Medio	Difícil

**Fuente:** (Ramírez, 2016, p.3); (Ortega, 2016, <https://openwebinars.net/blog/que-es-postcss-introduccion/>)

**Realizado por:** Edwin Hernández, 2019

Después del análisis realizado de los pre-compiladores, se optó por el pre-compilador Sass para el desarrollo de la aplicación web debido a que la configuración resulta más entendible al momento de su utilización, teniendo un apoyo de la comunidad oficial para los desarrolladores y parámetros favorables con respecto a los demás precompiladores **Tabla 8-1**.

## 1.9. Precompilador Sass

Sass es un precompilador que hace del CSS algo realmente entretenido de escribir, no solo para diseñadores sino también para ingenieros con experiencia (Blanch, 2018, <https://www.arsys.es/blog/programacion/que-es-sass/>).

Es un metalenguaje que se usa de forma más estructural y limpia, además de añadir una sintaxis sencilla, atractiva y compatible con la de CSS 3 y todas las que vengan en adelante, al tiempo que da ciertas características que son de gran ayuda a la hora de crear hojas de estilo (Miguel Angel Alvarez 2017, <https://desarrolloweb.com/articulos/que-es-sass-usar-sass.html>).

### 1.9.1. Características

El precompilador Sass posee varias características que se puede diferenciar de los demás preprocesadores de acuerdo a (Chambi, 2019, p.3):

- Compatible al 100% con css3.
- Permite el uso de variables, estilos y Mixins.
- Incluye el uso de elementos básicos de programación como son las librerías.
- Se puede incluir numerosas funciones para la manipulación de colores y otros valores.
- Genera archivos css bien formateados y permite su configuración.

### 1.9.2. *Ventajas y desventajas de Sass*

Las ventajas y desventajas del precompilador Sass se encuentran en la **Tabla 9-1**, la cual se muestra a continuación:

**Tabla 9-1:** Ventajas y desventajas de Sass

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none"> <li>• Se puede anidar los selectores para no repetir, teniendo un archivo más legible.</li> <li>• Uso de variables y funciones, ayudando a escribir menos código y evitar que se repita.</li> <li>• Hay un número infinito de marcos construidos.</li> <li>• Compatible con todas las versiones de css.</li> <li>• Es activamente apoyada y desarrollada por cientos de desarrolladores.</li> </ul>	<ul style="list-style-type: none"> <li>• Debe aprenderse a usarse.</li> <li>• Puede llevar a algunas malas prácticas.</li> <li>• Repite código usando Mixins.</li> </ul>

**Fuente:** (Xalambri, 2014, [https://www.academia.edu/35937550/SASS\\_Manual\\_Javier\\_Guede](https://www.academia.edu/35937550/SASS_Manual_Javier_Guede))

**Realizado por:** Edwin Hernández, 2019.

### 1.10. Lenguaje de programación

La programación es lograr que una máquina haga lo que el ser humano le ordene mediante el uso de ciertas instrucciones dadas, esperando que el ordenador pueda reconocer y cumplir con dichas tareas asignadas (Almagro, 2011, p.2).

Los lenguajes de programación pueden ser agrupados en dos clases: el lenguaje de bajo nivel que solo es entendido por la máquina y no por el ser humano y el lenguaje de alto nivel que son instrucciones que pueden ser comprendidas por el ser humano (Barber y Ferrís, 2004, p. 6).

Ejemplos del lenguaje de bajo nivel son los siguientes: fortran, cobol entre otros; por otro lado los lenguajes de alto nivel son: Php, Java, Ruby, Python, etc (Celaya, 2010, p. 6).

Existe varios lenguajes de programación en la actualidad que son los más utilizados como son Php, Java y Ruby (Almagro, 2011, p.2).

### **Php**

Es un lenguaje de programación para crear contenido html, ejecutado de diferentes manera en el servidor web, por líneas de comandos, mediante un cliente GUI; puede ser utilizado en cualquier plataforma o sistema operativo, tiene soporte para una variedad de base de datos y una comunidad de desarrolladores ya establecida en el mercado (Arce, 2018, p.3).

### **Java**

Es un lenguaje de programación que funciona en distintos tipos de procesadores, a partir de definir las clases creando los objetos que se conoce como instancias de la clase, la cual consta de propiedades o también llamados campos y métodos para manipular el estado de un objeto (Salinas, 2004, p.2).

### **Ruby**

Es un lenguaje de programación orientado a objetos; combina sintaxis inspirada en Python y Perl, el lenguaje soporta herencia con enlace dinámico, Mixins y métodos singleton (Bautista, 2011, <http://aplicaciones-web-lenguajes-programaci.blogspot.com/2011/12/ruby.html>).

#### ***1.10.1. Comparativa de los lenguajes de programación***

En la **Tabla 10-1**, se muestra la comparativa entre lenguajes de programación más usados para aplicaciones web; con los parámetros siguientes: sistema operativo, soporte es asistencia técnica, lenguaje interpretado no necesita ser preprocesado mediante un compilador, open source es de licencia libre, comunidad sitio oficial, programación mediante el cual es diseñado, rendimiento es la velocidad de realizar una tarea y en base a que lenguaje fue desarrollado.

**Tabla 10-1:** Comparativa de los lenguajes de programación

<b>PARÁMETROS</b>	<b>PHP</b>	<b>JAVA</b>	<b>RUBY</b>
Sistema Operativo	Windows	Windows	Linux
Soporte	SI	SI	NO

Lenguaje interpretado	SI	SI	SI
Open Source	SI	SI	SI
Comunidad	Usada por millones de desarrolladores	Tiene comunidad oficial	Falta en la comunidad oficial
Programación	Orientada a objetos	Orientada a objetos	Orientada a objetos
Rendimiento	Más eficiente	Eficiente	Poco más lento
Desarrollo	Fue programado en C y C++	Fue programado en C y C++	Fue programado en C

**Fuente:** (Ipiña, 2006, p.3); (Miró, 2016, <https://www.deustoformacion.com/blog/programacion-diseno-web/7-caracteristicas-lenguaje-php-que-lo-convierten-uno-mas-potentes>)

**Realizado por:** Edwin Hernández, 2019

Por la información recabada **Tabla 10-1**, se ha optado por PHP como lenguaje de programación para desarrollar la aplicación web; porque tiene suficiente información gracias a la comunidad de desarrolladores y la compatibilidad con cualquier sistema gestor de base de datos. Además de ser multiplataforma facilitando la interacción con el usuario del sistema y el acceso a la información.

### **1.11. Lenguaje de programación PHP**

PHP es un lenguaje embebido en páginas HTML y se ejecuta en los servidores. Inicio como una modificación a Perl escrita por Rasmus Lerdorf a finales de 1994. Su primer uso fue el de mantener un control sobre quien visitaba su curriculum en su web (Cobo et al., 2005, p.23).

Con PHP se puede hacer cualquier cosa que podemos realizar con un script CGI (Common Gateway Interface), como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. Permite la interacción con el visitante (González, 2016, p. 6).

#### **1.11.1. Características del lenguaje PHP**

El lenguaje de programación PHP posee ciertas características que se los puede diferenciar de otros lenguaje de programación utilizados de acuerdo a (Mariño, 2008, p. 10-14) , y son:

- Las aplicaciones desarrolladas en php son dinámicas para la web, frecuentes en los sitios comerciales (e-commerce), donde el contenido visualizado se genera de la información alcanzada en una base de datos u otra fuente externa.
- Php tiene soporte para potentes bases de datos siendo estas MySql, Oracle, PostgreSQL, entre otras.
- PHP también ofrece la integración con las varias bibliotecas externas, que permiten al desarrollador realizar cualquier actividad desde generar documentos en pdf hasta analizar código XML.
- PHP ofrece una solución simple y universal para las paginaciones dinámicas en la web de fácil programación.
- Debido a su amplia distribución php tiene soporte por una gran comunidad de desarrolladores.

### ***1.11.2. Ventajas y desventajas del lenguaje de programación PHP***

Las ventajas y desventajas del lenguaje de programación PHP se encuentran en la **Tabla 11-1**, las cuales se muestran a continuación:

**Tabla 11-1:** Ventajas y desventajas de Php

<b>VENTAJAS</b>	<b>DESVENTAJAS</b>
<ul style="list-style-type: none"> <li>• Es un lenguaje que se aprende rápido.</li> <li>• Se utiliza programación orientada a objetos para a herencia y las clases.</li> <li>• No se rige a un sistema operativo.</li> <li>• No requiere la especificación de las variables a utilizar.</li> </ul>	<ul style="list-style-type: none"> <li>• Necesita tener instalado previamente un servidor web.</li> <li>• Las peticiones del cliente pueden generar lentitud en las respuestas.</li> </ul>

**Fuente:** (García, 2010, pp. 9-10)

**Realizado por:** Edwin Hernández, 2019

### **1.12. Base de Datos**

Se utiliza la base de datos como una herramienta para almacenar datos y su posterior manipulación, los datos pueden ser de diferentes tipos; en las hojas de cálculo que muchas veces son utilizadas se pierde la fiabilidad e integridad de información por el alto número de datos que deben ser almacenados con el tiempo generando demora y perdido de información relevante, por lo que resulta más sencillo la migración a una base de datos computarizada donde la

información se encuentre ordenada en tablas mejorando de esta manera la eficiencia de los mismos (Pérez, 2011, <http://glo.org.mx/>).

Se entiende por base de datos un conjunto de datos no redundantes, almacenados en un soporte informático, organizado de forma independiente de su utilización y accesible simultáneamente por distintos usuarios y aplicaciones (Gómez, 2005, p.16).

### 1.12.1. Características

Las principales características que una base de datos posee son las siguientes de acuerdo con (Rubio, 2014, pp. 1-5):

- **Independencia de los datos:** los datos que posee la base de datos no dependen de cierto programa es decir que los datos pueden ser utilizados por otras aplicaciones.
- **Seguridad de acceso y auditoria:** a medida que se haga el registro de usuarios deberá crear una bitácora con todas las actividades en la BD para un control de seguridad de los datos.
- **Reducción de la redundancia:** en lo posible evitar la duplicación de los datos para conseguir mayor aprovechamiento de espacio e impedir que exista inconsistencias entre los datos.
- **Integridad de los datos:** valores reales que se almacenan en la base de datos para la utilización en las diversas estructuras de la aplicación.

### 1.12.2. Ventajas y desventajas de las bases de datos

Las ventajas y desventajas que existen en las bases de datos más utilizadas del mercado se encuentran en la **Tabla 12-1**, la cual se muestra a continuación:

**Tabla 12-1:** Ventajas y desventajas de las bases de datos

Ventajas	Desventajas
<ul style="list-style-type: none"> <li>• Evita la duplicidad de registros (Sánchez, 2004, <a href="https://www.mysql.com/">https://www.mysql.com/</a>).</li> <li>• Garantiza la integridad de los datos.(Oracle, 2007, p.4)</li> </ul>	<ul style="list-style-type: none"> <li>• El hardware y software para un correcto funcionamiento de la BD son costosas (Cibertareas 2019, <a href="https://cibertareas.info/ventajas-y-">https://cibertareas.info/ventajas-y-</a></li> </ul>

<ul style="list-style-type: none"> <li>• Permite compartir información de forma simultánea con otras base de datos (Sánchez, 2004, <a href="https://www.mysql.com/">https://www.mysql.com/</a>)</li> <li>• Facilita la estandarización de los procesos (Sarría, 2007, p. 4)</li> <li>• La información siempre se encuentra disponible para todos los usuarios (Sánchez, 2004, <a href="https://www.mysql.com/">https://www.mysql.com/</a>).</li> </ul>	<p>desventajas-de-las-bases-de-datos-herramientas-informaticas-1.html).</p> <ul style="list-style-type: none"> <li>• Expuesta a fallos en caso de existir alguna descarga eléctrica (Cibertareas 2019, <a href="https://cibertareas.info/ventajas-y-desventajas-de-las-bases-de-datos-herramientas-informaticas-1.html">https://cibertareas.info/ventajas-y-desventajas-de-las-bases-de-datos-herramientas-informaticas-1.html</a>).</li> <li>• Entre más grande sea la BD mayor será la capacidad de disco duro y memoria RAM para su correcto funcionamiento (Cibertareas 2019, <a href="https://cibertareas.info/ventajas-y-desventajas-de-las-bases-de-datos-herramientas-informaticas-1.html">https://cibertareas.info/ventajas-y-desventajas-de-las-bases-de-datos-herramientas-informaticas-1.html</a>).</li> </ul>
--	--

Realizado por: Edwin Hernández, 2019

Las bases de datos más utilizadas en el mercado según los desarrolladores de acuerdo a (Diarlu, 2019, <https://www.diarlu.com/gestores-bases-datos/>) son las siguientes:

### SQL Server

El sistema gestor de base de datos SQL Server no tiene dificultad en trabaja con diferentes sistemas operativos, bajo la declaración de instrucciones SQL permitiendo la disponibilidad en diferentes tareas que son demandadas (Sheldon, 2006, p.10).

### Oracle Database

Oracle es un sistema gestor de base de datos de lenguaje bien marcado de instrucciones que no se pueden ejecutar en otros SGBD, compatible en sus versiones con los principales sistemas operativos actuales en el mercado (Oracle, 2007, p. 4).

### MySql

Tiene como base el software libre, ofrece un gran rendimiento y la facilidad para crear una base de datos; comúnmente utilizado en conjunto con el lenguaje de programación PHP para el desarrollo web (Sánchez, 2004, <https://www.mysql.com/>).

### PostgreSQL

Es un sistema gestor de base de datos relacional, proporcionando fiabilidad e integridad en el almacenamiento de los datos mediante procesos entre los que están: disparadores, transacciones, etc; lo cual garantiza su buena manipulación (Sarría, 2007, p. 2).

### 1.12.3. Comparativa de las bases de datos

Las base de datos más utilizadas para el desarrollo de sistemas web de acuerdo con (Diarlu, 2019, <https://www.diarlu.com/gestores-bases-datos/>) a continuación, en la **Tabla 13-1**; con los parámetros siguientes: gratuito, multiplataforma puede funcionar en diversas plataformas, confiable es la capacidad de realizar su función de la manera prevista y escalable propiedad de aumentar la capacidad de trabajo, rendimiento velocidad de realizar una tarea, transacciones es un conjunto de órdenes que se ejecutan formando una unidad de trabajo y procedimientos almacenados es un programa almacenado físicamente en una base de datos.

**Tabla 13-1:** Comparativa de las bases de datos

PARÁMETROS	SQL SERVER	ORACLE DATABASE	MYSQL	POSTGRESQL
Gratuito	Si	No	Si	Si
Multiplataforma	Si	Si	Si	Si
Confiable y escalable	-	Si	Si	-
Rendimiento	Medio	Muy Alto	Alto	Medio
Transacciones	-	Presenta cuadros regulares y ayuda de operaciones	InnoDB	InnoDB
Procedimientos almacenados	Si	Si	No	Si

**Fuente:** (Mejia, 2014, <https://github.com/tatianamejia/Bases-de-Datos/wiki/Bases-de-Datos>; Florencio, 2017, <https://www.cursosgis.com/comparativa-de-los-principales-sistemas-gestores-de-bases-de-datos-sgbd/>)

**Realizado por:** Edwin Hernández 2019

Luego del análisis de los principales gestores de base de datos se destaca MySQL debido a que dispone de una alta gama de información siendo fácil de aprender y combinable con el lenguaje de programación PHP para el desarrollo de aplicaciones web, se ha decidido en este trabajo de titulación utilizará este gestor de base de datos.

### 1.13. Sistema Gestor de base de datos MySQL

MySQL es de tipo relacional de código abierto, cuenta con 16 años de experiencia aproximadamente en el desarrollo web proveyendo de fiabilidad e integridad en los datos; pudiéndose ejecutar en cualquier sistema operativo, almacenando gran cantidad de datos grandes entre los cuales están imágenes, videos, audio, etc (Sánchez, 2004, <https://www.mysql.com/>).

MySQL es un Sistema Gestor de base de datos orientado a objetos, el cual es reconocido por sus componentes; tiene licencia BSD (distribución de software Berkeley) lo cual permite su uso y poder mejorarlo por la experiencia adquirida (Sánchez, 2004, <https://www.mysql.com/>).

### **1.13.1. Características de SGBD MySQL**

Entre las principales características de MySQL se encuentra las siguientes:

- Ser de código abierto y gratuito (Lockhart, 1996, p.24).
- Permite ejecutarse en los principales sistemas operativos (Lockhart, 1996, p.24).
- Las transacciones mediante ACID (Atomic, Consistent Isolated, Durable): Para construir aplicaciones más seguras mediante el bloqueo por registro (Serrano, 2008, p. 41).
- El almacenamiento ofrece diferentes velocidades de operación, soporte físico, capacidad, distribución geográfica y transacciones (Serrano, 2008, p. 41).
- Mediante una arquitectura cliente-servidor él cual permite la comunicación entre sí para un mejor rendimiento del sistema (Serrano, 2008, p. 41).

## **1.14. Metodología de desarrollo de software**

La metodología es utilizada para estructurar, planificar y controlar el proceso de desarrollo en sistemas informáticos. Teniendo métodos que han sido desarrollados dependiendo de sus fortalezas y debilidades (Ruiz, 2017, p. 6).

Se origina para el desarrollo de manera escalar funcional de sistema de negocios, la idea principal es desarrollar los sistemas de manera estructurada y metódica. Reiterando cada una de las etapas del ciclo de vida del software (Espinoza-Meza 2013, p.4).

Teniendo como objetivo presentar un conjunto de técnicas tradicionales y modernas de modelado de sistemas que permitan desarrollar software de calidad, incluyendo heurísticas de construcción y criterios de comparación de modelos de sistemas (Espinoza-Meza 2013, pp. 6-7).

## **RAD**

Rapid Application Development. El método comprende el desarrollo interactivo, que permite la construcción de prototipos y el uso de utilidades CASE en poco tiempo normalmente ente 60 y 90 días, frecuentemente con algunas concesiones (Pressman, 2005, p. 42).

## SCRUM

La metodología Scrum se basa en un conjunto de buenas prácticas para el eficiente trabajo, obteniendo mejores resultados y las continuas entregas parciales para no tener atrasos; terminación con éxito del producto software teniendo como prioridad los requerimientos del cliente (Álvarez, 2016, p.12).

## XP

Es una metodología ágil y flexible utilizada para la gestión de proyectos, se centra en potenciar las relaciones interpersonales del equipo de desarrollo haciendo cumplir estrictamente las reglas que se centran en las necesidades del cliente para lograr un producto de buena calidad en poco tiempo (López, 2013, p. 2).

### 1.14.1. Comparativa de metodologías de software

En la **Tabla 14-1**, se describe una serie de parámetros que ayudarán a decidir qué metodología aplicar; con los parámetros siguientes: tamaño de equipo, sprint backlog es el listado de tareas de una iteración, roles función que cumple alguien, entregas parciales, cambios y proyectos.

**Tabla 14-1:** Comparativa de metodología de software

PARÁMETROS	RAD	SCRUM	XP
Tamaño de equipo	Sin prescripción	Máximo 10	Sin prescripción
Sprint Backlog	No permite añadir tareas	Rígida no permite modificaciones posteriores a lo entregable	Permite añadir tareas
Roles	No existe	Scrum máster, producto owner y el equipo de desarrollo	Tracker, customer, programmer, coach, manager, tester
Entregas parciales	Sus entregas se lo realizan lo más pronto posible	Se realizan cada 2 o 4 semanas	Se realizan de 1 a 3 semanas lo más rápido posible.
Cambios	No acepta cambios durante el proceso de desarrollo.	Toma en cuenta la administración del proyecto en desarrollo	Centra principalmente en la programación o desarrollo del proyecto
Proyectos	Proyectos pequeños	Proyectos bien definidos los requerimientos	Para proyectos con requisitos indeterminados y multi

			variantes
--	--	--	-----------

Fuente: (Hmd Project Managers, 2015, <https://uv-mdap.com/blog/metodologias-agiles/>)

Realizado por: Edwin Hernández 2019

Posterior al análisis de las principales metodologías se utilizará la metodología ágil Scrum misma que permite realizar entregas cada 2 semanas para que el cliente conozca cómo se va desarrollando la aplicación el mismo que también compromete para que siga interactuando con el desarrollador e ir aclarando dudas sea este el caso.

## 1.15. Metodología SCRUM

La metodología ágil Scrum ayuda para gestionar el desarrollo de software, teniendo como principal objetivo es maximizar el retorno de la inversión para su empresa (ROI). Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, autogestión e innovación (Gallego, 2010, pp. 35-37).

Con la metodología Scrum el cliente se entusiasma y se compromete con el proyecto dado que lo ve crecer iteración a iteración. Así mismo, le permite en cualquier momento realinear el software con los objetivos de su empresa, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración sin ningún problema. Esta metódica de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades (Ken, 2013, pp. 4-5).

### 1.15.1. Roles de la metodología SCRUM

En Scrum, el equipo se focaliza en construir software de calidad. La gestión de un proyecto Scrum se centra en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo (Álvarez, 2013, p. 7).

El equipo Scrum está formado por los siguientes roles:

- **Scrum máster:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner para maximizar el ROI (Álvarez, 2013, p. 11).

- **Product Owner (PO):** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio y él es responsable del ROI del proyecto (entregar un valor superior al dinero invertido). Traslada la visión del proyecto al equipo, formaliza las prestaciones en historias a incorporar en el Product Backlog y las Re prioriza de forma regular (Palacio, 2015, <http://streetsofdublin.com/>).
- **Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint (Palacio, 2015, <http://streetsofdublin.com/>)

### *1.15.2. Proceso para el desarrollo del software*

Para el desarrollo del software se lo realiza de manera iterativa e incremental, cada iteración denominada Sprint tiene una duración de 2 semanas, en las cuales se ven los resultados y la nueva versión del sistema desarrollado. Así también se ajusta funcionalidades y se van añadiendo nuevas prestaciones en el avance del proyecto (Martín, 2018, <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>).

- **Product Backlog:** Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares (Martín, 2018, <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>).
- **Sprint Planning:** Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir (Martín, 2018, <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>).
- **Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo (Menzinsky, 2016, <http://streetsofdublin.com/>).
- **Sprint Backlog:** Lista de las tareas necesarias para llevar a cabo las historias del sprint (Menzinsky, 2016, <http://streetsofdublin.com/>).

- **Daily sprint meeting:** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos (Menzinsky, 2016, <http://streetsofdublin.com/>).
- **Demo y retrospectiva:** Reunión que se celebra al final del sprint y en la que el equipo presenta las historias conseguidas mediante una demostración del producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos (Peralta, 2003, p. 9).

### *1.15.3. Beneficios de la metodología SCRUM*

- **Cumplimiento de expectativas:** El cliente establece sus expectativas indicando el valor que le aporta cada requisito / historia del proyecto, el equipo lo estima y con esta información el Product Owner establece su prioridad. De manera regular, en las demos de Sprint el Product Owner comprueba que efectivamente los requisitos se han cumplido y transmite el feedback al equipo (Peralta, 2003, p. 8).
- **Flexibilidad a cambios:** Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado. La metodología está diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos (Peralta, 2003, p. 8).
- **Reducción del Time to Market:** El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo (Teralco, 2002, p. 7).
- **Mayor calidad del software:** Se necesita un ritmo de trabajo por las funcionalidades de cada una de las iteraciones planteadas, mejorando de esta manera la calidad de software desarrollado (Teralco, 2002, p. 8).
- **Mayor productividad:** Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse (Teralco, 2002, p. 8).
- **Maximiza el retorno de la inversión (ROI):** Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión (Teralco, 2002, p. 8).

- **Predicciones de tiempos:** Mediante esta metodología se conoce la velocidad media del equipo por sprint (los llamados puntos historia), con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está en el Backlog (Deemer, 2012, <http://www.goodagile.com/>).
- **Reducción de riesgos:** Se debe realizar primero las funcionalidades que tienen mayor prioridad para tener una apreciación de la velocidad de trabajo del equipo de desarrollo, permitiendo tener un control en caso de complicaciones posteriores (Deemer, 2012, <http://www.goodagile.com/>).

## 1.16. Estándar de evaluación del software

En la evaluación de software es muy importante acatar modelos y estándares actuales de calidad, que detallan las características mediante las cuales se hace la evaluación, las métricas de calidad y las métricas técnicas de calidad de software que son los indicadores (Peña, 2011, p. 3).

Es necesario tener en cuenta las escalas de medición cualitativa o cuantitativa y las pruebas de software que se ejecutan para verificar la calidad del producto o proceso (Peña, 2011, p. 4).

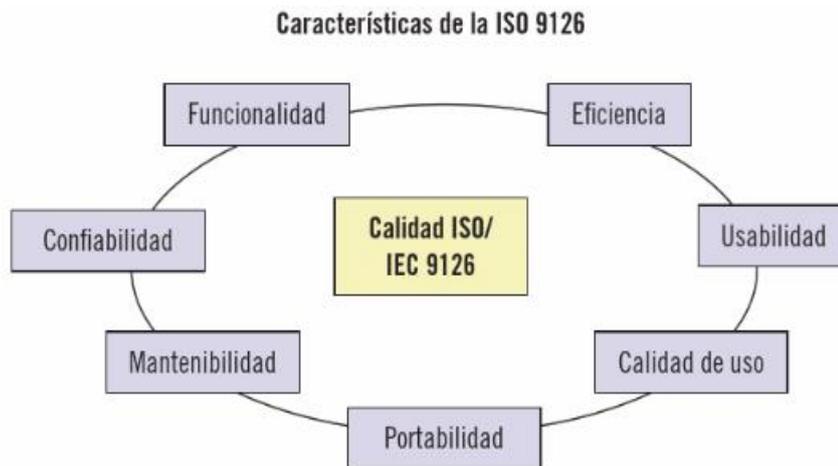
Ofreciendo normas y parámetros con pasos específicos para la creación de proyectos informáticos. La calidad de software es fundamental para las empresas y su evaluación se hace pertinente para que se cumplan los propósitos que se quieren lograr con la ayuda de esos productos software (Marqués, 2016, p. 11).

### 1.16.1. Norma ISO 9126

La norma ISO 9126 tiene como propósito la evaluación de la calidad de software, consta de cuatro partes principales las que son: modelo de calidad, métricas internas y externas y la calidad de uso (Figuroa, 2004, <http://www.monografias.com/trabajos5/>).

El modelo de calidad cuenta con características que deben existir en un software formado por dos partes calidad interna y calidad externa (Figuroa, 2004, <http://www.monografias.com/trabajos5/>).

En la **Figura 4-1**, se hace referencia a las características de la calidad de la ISO/IEC 9126 la cual se muestra a continuación:



**Figura 4-1:** Características de la ISO 9126  
**Fuente:** (Durán, 2015, p.29)

### ***1.16.2. Factores de calidad***

Los factores de la calidad de la Norma ISO 9126 son las siguientes (Raúl et al., 2016, pp. 31-35) :

**Funcionalidad:** capacidad que posee el software para satisfacer las necesidades del cliente. Sus subcaracterísticas son: adecuación, exactitud, interoperabilidad, seguridad y cumplimiento funcional.

**Fiabilidad:** conjunto de atributos que permiten mantener el nivel de prestación ante un periodo ya establecido, consta de 4 subcaracterísticas: madurez, recuperabilidad, tolerancia a fallos y cumplimiento de fiabilidad.

**Usabilidad:** conjunto de propiedades que le permiten interactuar con todos los roles para poder trabajar y gestionar de una manera correcta, la misma que se subdivide en subcaracterísticas siendo estas: aprendizaje, comprensión, operatividad y atractividad.

**Eficiencia:** aquella que se refiere a la valoración y cuantificación entre la cantidad de recursos que ofrece la aplicación. Sus subcaracterísticas son: comportamiento de tiempos y utilización de recursos.

**Mantenibilidad:** cuantifica la facilidad de la aplicación para ser ampliada alterada o depurada, sus subcaracterísticas son: estabilidad, facilidad de análisis, cambio y pruebas.

**Portabilidad:** capacidad que dispone la aplicación para ser trasladado de una a otra plataforma, cuenta con las siguientes subcaracterísticas: capacidad de instalación y reemplazamiento.

**Calidad de uso:** encargada principalmente de la seguridad de la aplicación y aprobación por parte del cliente del software desarrollado. Sus subcaracterísticas son: eficacia, productividad, seguridad y satisfacción.

### ***1.16.3. Eficiencia de la Norma Iso 9126***

Tiene una relación entre el nivel de funcionamiento del software con respecto a los recursos utilizados en un tiempo determinado, las subcaracterísticas de presentan a continuación:

- **Comportamiento de tiempos.** – Son los tiempos que se esperan al realizar ciertas funciones bajo determinadas condiciones establecidas previamente (BORBÓN, 2013, <http://actividadreconocimiento-301569-8.blogspot.com/2013/03/norma-de-evaluacion-isoiec-9126.html>).
- **Utilización de recursos:** Se utiliza los recursos que posee el software mediante los cuales realiza sus funciones normalmente, bajo ciertas condiciones establecidas (Durán, 2015, p. 34)

Para el análisis del desarrollo de la aplicación web se utilizará la norma ISO 9126. La métrica para analizar el presente trabajo de titulación es la eficiencia; enfocándose en la subcaracterísticas de comportamiento de tiempos de los procesos de registro de producto, proveedores, clientes y pedidos, teniendo en cuenta que mientras menor sea el tiempo de respuesta mejor será para el cliente.

## CAPÍTULO II

### 2. MARCO METODOLÓGICO

#### 2.1. Diseño de la investigación

##### 2.1.1. *Tipo de investigación*

Para este trabajo de titulación se utiliza la investigación aplicada, mediante la cual se va aplicar los conocimientos teóricos y prácticos adquiridos durante los años de estudio, lo que contribuirá en el desarrollo de la aplicación web y el almacenamiento de datos mediante un sistema de gestor de base de datos, satisfaciendo de esta manera a la Empresa con un producto eficiente y funcional.

##### 2.1.2. *Métodos de investigación*

Los métodos de investigación que se utiliza para este trabajo de titulación son el método de análisis-síntesis y el método inductivo-deductivo.

- **Método de Análisis-Síntesis**

Mediante el análisis se estudia como la Empresa El Chagra lleva la información de los clientes, productos, proveedores y pedidos; con la prioridad de entender la gestiona estos procesos.

Con la síntesis se reunió la información requerida de las tecnologías y herramientas a utilizar para desarrollar la aplicación web, que posterior se plasmará en el marco teórico referencial.

- **Método Inductivo-Deductivo**

Mediante el método inductivo se inicia de lo particular a lo general, analizando el funcionamiento de la aplicación web para llegar a conclusiones que después será puesto en los resultados.

Con el método deductivo se parte de lo general a lo particular, para realizar las recomendaciones teniendo en cuenta las conclusiones que se tuvo del método inductivo.

### **2.1.3. Técnicas de investigación**

Mediante la utilización de las técnicas de entrevista, observación y revisión de documentación, descritas a continuación, se pudo reunir la información necesaria para el desarrollo de la aplicación web:

- Entrevista con el Ing. William Rolando Cayo Llumiluisa dueño de la Empresa “El Chagra” para obtener los requisitos de la aplicación web.
- Observación de los procesos necesarios para la toma de los tiempos mediante la utilización de un cronómetro.
- Revisión de documentación de la gestión de pedidos que se realiza de forma tradicional en la Empresa “El Chagra” y las diferentes fuentes bibliográficas para la realización del marco teórico referencial.

## **2.2. Determinación de los procesos que se manejan en la Empresa “El Chagra”**

El proceso de pedidos de los productos agrícolas de la Empresa “El Chagra” tiene como prioridad gestionar la respectiva información de cada uno de los productos que ofrece, de manera que toda esta información no se encuentra automatizada lo cual provoca demora en la gestión de pedidos de productos agrícolas por parte del usuario.

### **2.2.1. Diagrama de procesos no automatizado**

En este proceso intervienen cuatro actores principales el cliente, empleado, administrador y el proveedor; en tres procesos registro de cliente, realizar pedido y entrega del pedido.

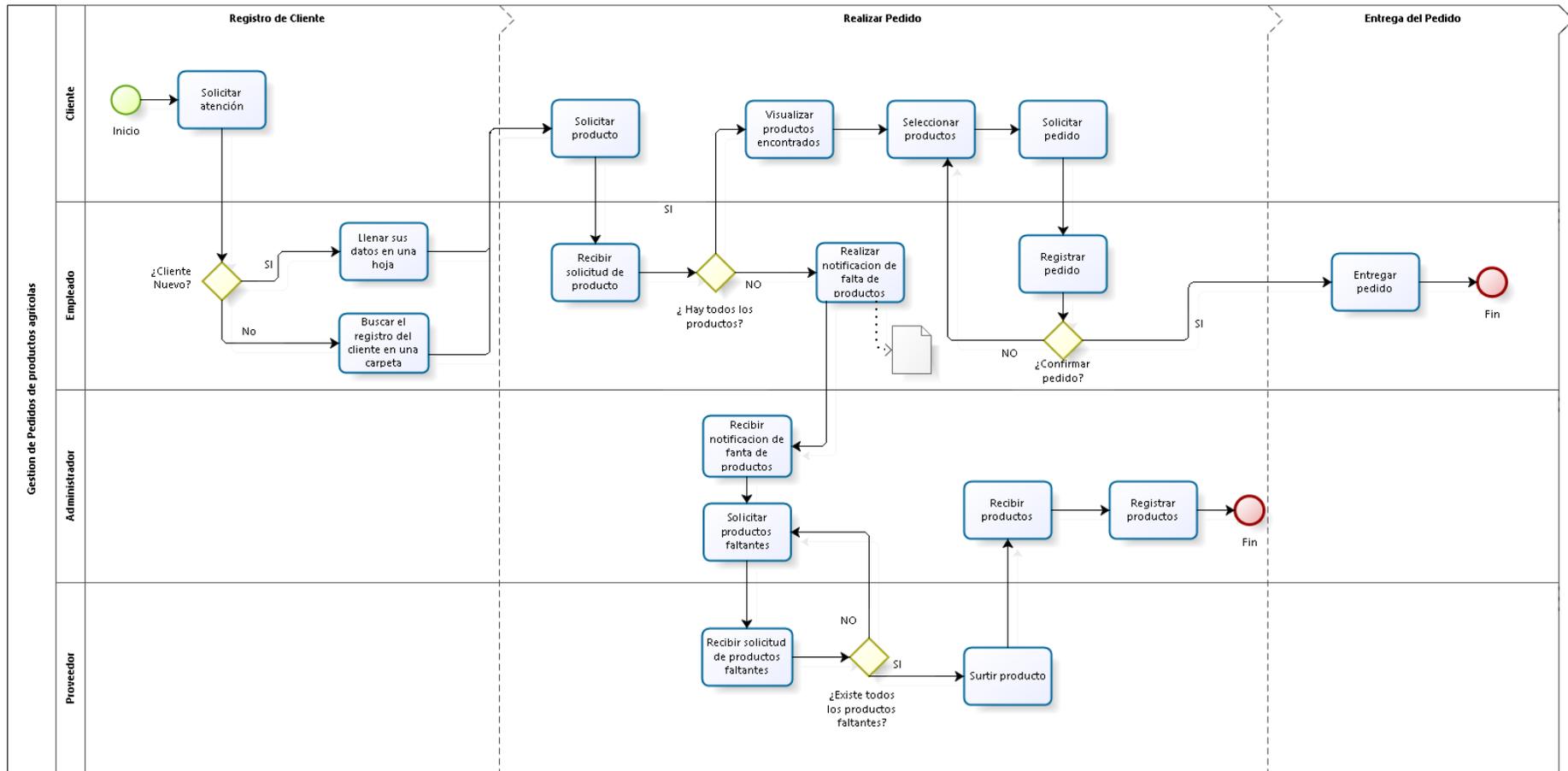
A continuación, se presenta la secuencia de actividades del proceso no automatizado de la gestión de pedidos de productos agrícolas:

#### **Secuencia de Actividades:**

1. Solicitar atención: El cliente llega a la Empresa “El Chagra” realizar el pedido de los productos.

2. Llenar sus datos en una hoja: El empleado solicita los datos del cliente para ser llenado en una hoja de ficha del cliente.
3. Buscar el registro del cliente en una carpeta: Posteriormente el empleado busca el registro del cliente en caso de no ser nuevo.
4. Solicitar producto: El cliente solicita el producto agrícola deseado.
5. Recibir solicitud de producto: El empleado recibe la solicitud de parte del cliente y confirma la existencia del producto.
6. Visualizar productos encontrados: El cliente visualiza los productos agrícolas que requiere para su pedido.
7. Seleccionar productos: El cliente selecciona los productos agrícolas existentes.
8. Solicitar pedido: Mediante una solicitud el cliente realiza el pedido al empleado.
9. Registrar pedido: El empleado confirma el pedido y lo registra para el posterior despacho de mercadería.
10. Entregar pedido: El empleado entrega el pedido de los productos agrícolas al cliente.
11. Realizar notificación de falta de productos: En caso que no existan los productos solicitados se procede a realizar una notificación para el Administrador para su posterior adquisición.
12. Recibir notificación de falta de productos: El administrador recibe la notificación de falta de productos en el stock.
13. Solicitar productos faltantes: El administrador solicita los productos faltantes al proveedor.
14. Recibir solicitud de productos faltantes: El proveedor recibe la solicitud de los productos faltantes y verifica si tiene lo necesario para abastecer a la empresa.
15. Surtir producto: El proveedor entrega el producto necesario en la Empresa.
16. Recibir productos: El administrador recibe el producto de parte del proveedor.
17. Registrar productos: El administrador registra el nuevo producto recibido.

En la **Figura 1-2**, se indica de forma gráfica los procesos no automatizados de la gestión de productos agrícolas de la Empresa “El Chagra”, los cuales mediante la herramienta Bizagi Modeler son representados a continuación:



**Figura 1-2:** Diagrama de procesos no automatizados de la gestión de pedidos  
Realizado por: Edwin Hernández, 2019

### **2.2.2. Diagrama de procesos automatizado**

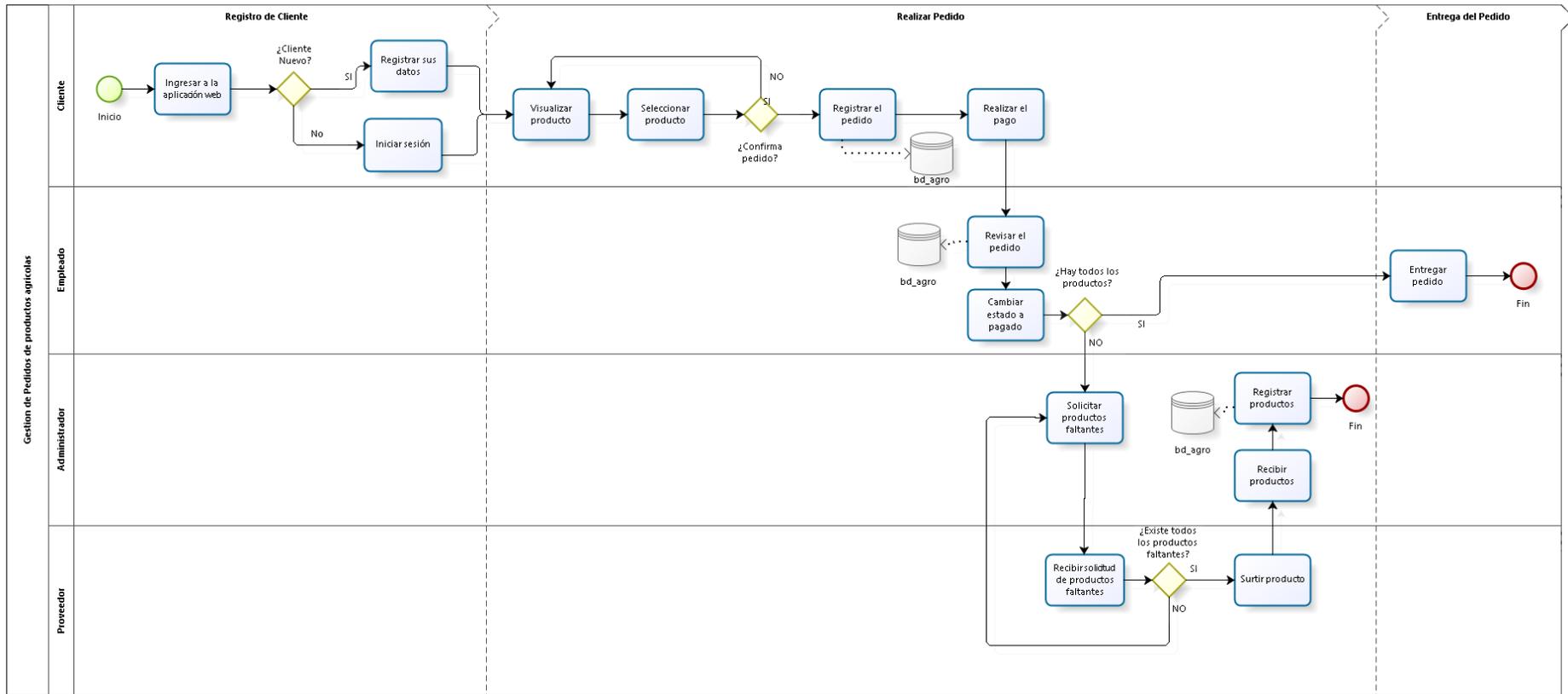
En el proceso automatizado interviniendo cuatro actores principales el cliente, empleado, administrador y el proveedor; en tres procesos registro de cliente, realizar pedido y entrega del pedido.

A continuación, se presenta la secuencia de actividades del proceso automatizado de la gestión de pedidos de productos agrícolas:

#### **Secuencia de Actividades:**

1. Ingresar a la aplicación web: El cliente ingresa a través de un navegador web a la aplicación de la Empresa.
2. Registrar sus datos: El cliente registra sus datos si es cliente nuevo ingresando al sistema.
3. Iniciar sesión: Caso contrario ingresa con sus respectivas credenciales (usuario y contraseña).
4. Visualizar productos: El cliente visualiza los productos depende de la categoría en la aplicación web.
5. Seleccionar producto: Posteriormente el cliente selecciona los productos agrícolas que necesita y confirma el pedido.
6. Registrar el pedido: El cliente confirma el pedido de los productos agrícolas que ha realizado se registra en la base de datos.
7. Realizar el pago: El cliente realiza el pago de los productos agrícolas requeridos.
8. Revisar el pedido: El empleado revisa el pedido que se ha realizado de los productos agrícolas y revisa si hay todos los productos.
9. Cambiar estado ha pagado: El empleado cambia el estado del pedido a pagado.
10. Entregar pedido: El empleado entrega el pedido al cliente.
11. Solicitar productos faltantes: En caso de faltar productos el administrador solicita los productos que faltan en la empresa.
12. Recibir solicitud de productos faltantes: El proveedor recibe la solicitud de los productos que faltan y revisa si tiene todo lo necesario.
13. Surtir producto: El proveedor entrega los productos a la empresa.
14. Recibir producto: El administrador recibe los productos del proveedor.
15. Registrar productos: El administrador registra los productos que son guardados en la base de datos.

En la **Figura 2-2**, se indica de forma gráfica los procesos automatizados de la gestión de productos agrícolas de la Empresa “El Chagra”, los mismos que se han sido plasmados mediante la herramienta Bizagi Modeler.



**Figura 2-2:** Diagrama de procesos Automatizado  
Realizado por: Edwin Hernández, 2019

## 2.3. Fase de planificación

En esta etapa de planificación se debe realizar un análisis para poder identificar el problema principal que manifiesta el cliente, definiendo de esta manera que es lo que requiere; manteniendo reuniones continuas e informando lo que se va a realizar en el proyecto mediante las actividades planteadas anteriormente.

### 2.3.1. Personas involucradas

Las personas involucradas en el desarrollo del proyecto se encuentran en la **Tabla 1-2**, que se visualiza a continuación:

**Tabla 1-2:** Personas involucradas en el desarrollo del proyecto

PERSONA	ROL	CONTACTO
Ing. Paúl Paguay	SCRUM Master	<a href="mailto:ppaguay@epoch.edu.ec">ppaguay@epoch.edu.ec</a>
Ing. Rolando Cayo	Product Owner	<a href="mailto:rolandocayo2@hotmail.com">rolandocayo2@hotmail.com</a>
Edwin Hernández	Development Team	<a href="mailto:edwin.berrones@epoch.edu.ec">edwin.berrones@epoch.edu.ec</a>

Realizado por: Edwin Hernández, 2019

### 2.3.2. Tipos y roles de usuario

Para cada usuario se tiene descripción, rol y responsabilidad que se muestra a continuación en la **Tabla 2-2**:

**Tabla 2-2:** Tipos de usuarios y roles

TIPO DE USUARIO	DESCRIPCIÓN	ROL	RESPONSABLE
Administrador	La administración de la aplicación web	Utilizar y manejar todas las funcionalidades del sistema.	Ing. Rolando Cayo
Clientes	Puede realizar el pedido	Realizar pedido de productos	Clientes

Realizado por: Edwin Hernández, 2019

### 2.3.3. Actividades realizadas en el proyecto

Las actividades que se realizaron para el desarrollo de la aplicación web se encuentran en la **Tabla 3-2**, la cual se muestra a continuación:

**Tabla 3-2:** Actividades realizadas en el proyecto

<b>INSTITUCIÓN</b>	<b>ACTIVIDAD</b>	<b>RESPONSABLE</b>
<b>EMPRESA</b> <b>“EL CHAGRA”</b>	Mediante reuniones continuas con el cliente se reunió la requerimentación necesaria para el desarrollo.	Development Team
	Diseñar la arquitectura de la aplicación web	Development Team
	Diseñar de la base de datos de la empresa.	Development Team
	Diseñar las diferentes interfaces de usuario.	Development Team
	Desarrollar los módulos de la aplicación web.	Development Team
	Realizar la respectiva documentación del manual técnico y manual de usuario.	Development Team

Realizado por: Edwin Hernández, 2019

#### **2.3.4. Product backlog**

Mediante el product backlog se expresa los diferentes requerimientos que el cliente desea, en reuniones continuas con el Ing. William Rolando Cayo Llumiluisa descritos en el **Anexo A**; posteriormente se define las historias técnicas e historias de usuario para el desarrollo de la aplicación web contando con un grado de prioridad cada una.

Para realizar la estimación se utiliza el método denominado T-Shirt (Talla de Camiseta), el cual sirve para definir la estimación de cada uno de los requerimientos, dentro de lo mencionado, las tallas XL, L, M, S y XS son las que se utiliza para dar una medida de duración a los requerimientos, en la **Tabla 4-2** se puede observar de una forma detallada las estimaciones definidas.

**Tabla 4-2: Método T-Shirt**

TALLA	PUNTOS ESTIMADOS	HORAS DE TRABAJO
XS	8	8
S	16	16
M	20	20
L	40	40
XL	80	80

Realizado por: Edwin Hernández, 2019

Además, se recalca un día de trabajo tiene 8 horas, por lo tanto 1 semana de trabajo equivaldrá a 40 puntos estimados, 2 semanas de trabajo equivaldrá a 80 puntos estimados.

Se presenta el Product Backlog en la **Tabla 5-2**, a continuación:

**Tabla 5-2: Product Backlog**

<b>HISTORIAS TÉCNICAS</b>			
ID	ENUNCIADO	PRIORIDAD	ESTIMACIÓN
HT-01	Entrevista y especificación de requerimientos para la aplicación web	Alta	80
HT-02	Definir el estándar de programación	Media	20
HT-03	Diseño de la arquitectura del sistema	Media	20
HT-04	Diseño e implementación de la base de datos	Alta	40
HT-05	Diseño de las interfaz de usuario	Media	20
HT-06	Manual de usuario	Alta	20
<b>HISTORIAS DE USUARIO</b>			
HU-01	Desarrollo de la interfaz del módulo administrador	Media	20
HU-02	Desarrollo de la interfaz del módulo cliente	Media	8
HU-03	Ingresar información de producto	Alta	40
HU-04	Actualizar información de producto	Media	20
HU-05	Eliminar información de productos	Media	16

HU-06	Buscar productos	Media	16
HU-07	Listar productos	Baja	8
HU-08	Ingresar información del cliente	Alta	20
HU-09	Ingresar categoría de producto	Media	16
HU-10	Modificar categoría de producto	Media	16
HU-11	Ingresar información del administrador	Alta	20
HU-12	Carrito de compras	Media	16
HU-13	Eliminar categoría de producto	Baja	8
HU-14	Buscar administrador	Media	16
HU-15	Listar administrador	Baja	8
HU-16	Ingresar información de proveedor	Alta	20
HU-17	Actualizar información de proveedor	Baja	20
HU-18	Eliminar información de proveedor	Baja	40
HU-19	Buscar proveedor	Media	16
HU-20	Listar proveedor	Media	16
HU-21	Ingresar información del pedido	Alta	40
HU-22	Actualizar información del pedido	Media	16
HU-23	Eliminar información del pedido	Baja	8
HU-24	Buscar pedido	Media	16
HU-25	Listar pedido	Media	40
HU-26	Reporte de pedidos	Media	16
HU-27	Reporte de productos	Media	16
HU-28	Reporte de proveedores	Baja	8
<b>TOTAL</b>			720

Realizado por: Edwin Hernández, 2019

Se determinó 28 historias de usuario y 06 historias técnicas, las cuales se procederán a ubicar en sprint.

### 2.3.5. *Sprint backlog*

Las historias técnicas e historias de usuario de cada Sprint tienen los siguientes datos la duración en días, las fechas de comienzo y final de la historia y el tiempo estimado que se va a demorar; teniendo en cuenta que cada Sprint tiene una duración de dos semanas para su desarrollo, se puede visualizar en la **Tabla 6-2**, a continuación:

**Tabla 6-2:** Sprint Backlog

NOMBRE TAREA	DURACIÓN	FECHAS		ESTIMACIÓN
		COMIENZO	FINAL	
<b>Fase de desarrollo</b>	<b>90 días</b>	<b>02/01/2019</b>	<b>07/05/2019</b>	
<b>Sprint 1</b>	<b>10 días</b>	<b>02/01/2019</b>	<b>15/01/2019</b>	
Entrevista y especificación de requerimientos para la aplicación web	10 días	02/01/2019	15/01/2019	80
<b>Sprint 2</b>	<b>10 días</b>	<b>16/01/2019</b>	<b>29/01/2019</b>	
Definir el estándar de programación	2 ½ días	16/01/2019	18/01/2019	80
Diseño de la arquitectura del sistema	2 ½ días	18/01/2019	22/01/2019	
Diseño e implementación de la base de datos	5 días	23/01/2019	29/01/2019	
<b>Sprint 3</b>	<b>10 días</b>	<b>30/01/2019</b>	<b>12/02/2019</b>	
Diseño de la interfaz de usuario	2 ½ días	30/01/2019	01/02/2019	80
Desarrollo de la interfaz del módulo cliente	2 ½ días	01/02/2019	05/02/2019	
Desarrollo de la interfaz del módulo administrador	5 días	06/02/2019	12/02/2019	
<b>Sprint 4</b>	<b>10 días</b>	<b>13/02/2019</b>	<b>26/02/2019</b>	
Ingresar información de producto	2 ½ días	13/02/2019	15/02/2019	
Actualizar información de producto	2 ½ días	15/02/2019	19/02/2019	
Buscar productos	2 días	20/02/2019	21/02/2019	80
Listar productos	2 días	22/02/2019	25/02/2019	
Eliminar información de productos	1 día	26/02/2019	26/02/2019	
<b>Sprint 5</b>	<b>10 días</b>	<b>27/02/2019</b>	<b>12/03/2019</b>	

Ingresar información del administrador	2 ½ días	27/02/2019	01/03/2019	80
Ingresar información del cliente	2 ½ días	01/03/2019	05/03/2019	
Ingresar categoría de producto	1 día	06/03/2019	06/03/2019	
Modificar categoría de producto	2 días	07/03/2019	08/03/2019	
Ingresar información de proveedor	2 días	11/03/2019	12/03/2019	
<b>Sprint 6</b>	<b>10 días</b>	<b>13/03/2019</b>	<b>26/03/2019</b>	
Eliminar información del administrador	2 ½ días	13/03/2019	15/03/2019	80
Carrito de compras	2 ½ días	15/03/2019	19/03/2019	
Buscar administrador	2 días	20/03/2019	21/03/2019	
Listar administrador	2 días	22/03/2019	25/03/2019	
Actualizar información de proveedor	1 día	26/03/2019	26/03/2019	
<b>Sprint 7</b>	<b>10 días</b>	<b>27/03/2019</b>	<b>09/04/2019</b>	
Eliminar información de proveedor	1 día	27/03/2019	27/03/2019	80
Buscar proveedor	5 días	28/03/2019	03/04/2019	
Listar proveedor	2 días	04/04/2019	05/04/2019	
Ingresar información del pedido	2 días	08/04/2019	09/04/2019	
<b>Sprint 8</b>	<b>10 días</b>	<b>10/04/2019</b>	<b>23/04/2019</b>	
Actualizar información del pedido	5 días	10/04/2019	16/04/2019	80
Eliminar información del pedido	2 días	17/04/2019	18/04/2019	
Buscar pedido	1 día	19/04/2019	19/04/2019	
Listar pedido	2 días	22/04/2019	23/04/2019	
<b>Sprint 9</b>	<b>10 días</b>	<b>24/04/2019</b>	<b>07/05/2019</b>	
Reporte de pedidos	5 días	24/04/2019	30/04/2019	80
Reporte de productos	2 días	01/05/2019	02/05/2019	
Reporte de proveedores	2 días	03/05/2019	06/05/2019	
Manual de usuario	1 día	07/05/2019	07/05/2019	

Realizado por: Edwin Hernández 2019

Se desarrolló la aplicación web en 9 sprint comenzando el día **martes 02/01/2019** y finalizando el día **martes 07/05/2019**, con un total de 720 horas.

### 2.3.6. Reuniones SCRUM

Mediante las reuniones establecidas entre el desarrollador y el cliente se especifica los diferentes requerimientos de la aplicación web desarrollada, cumpliendo con lo que necesita el cliente y las expectativas de visualizar cada iteración un avance con el desarrollo del proyecto; pudiendo expresar los cambios que se necesiten para la terminación con éxito.

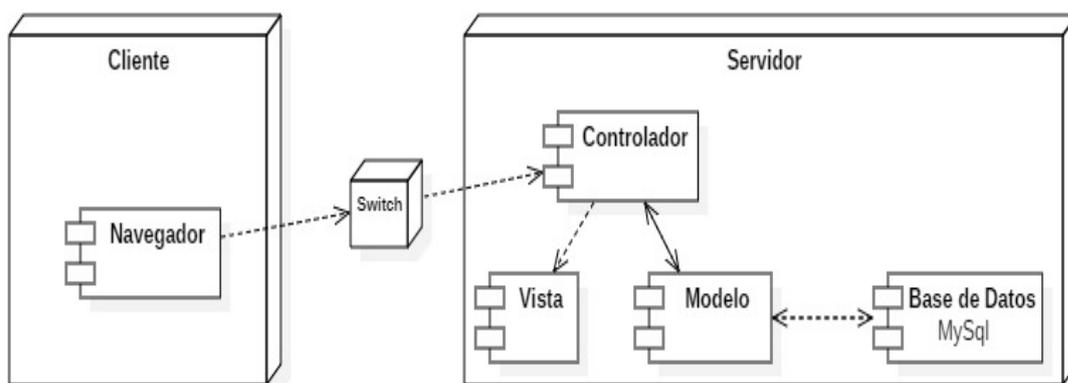
Se realizaron reuniones de inicio y fin de cada sprint, el desarrollador en la reunión de inicio expreso las funcionalidades que se va a realizar en el desarrollo de la aplicación web mientras que en la reunión de fin para mostrar los avances realizados para la culminación exitosa del proyecto.

## 2.4. Fase de desarrollo

### 2.4.1. Arquitectura del sistema

Para la arquitectura de la aplicación web se emplea la arquitectura Cliente-Servidor con el acompañamiento del patrón de diseño MVC (Modelo-Vista-Controlador), la cual implementa la capa denominada Vista donde se encuentra la interfaz del administrador y cliente; permitiendo ingresar a través de un navegador web, la capa Modelo es la que permite acceder a los datos de la base de datos generada con el gestor MySQL y la capa Controlador define las respectivas instrucciones para el negocio.

La arquitectura de la aplicación web se muestra en la **Figura 3-2**, a continuación:



**Figura 3-2:** Arquitectura de la aplicación web  
Realizado por: Edwin Hernández, 2019

### 2.4.2. Estándar de codificación

El estándar de codificación sirve para tener un conjunto claro del código que está siendo generado y su fácil comprensión al momento de la integración entre funcionalidades, teniendo en claro cómo se ha definido las diferentes variables así también las clases, atributos y métodos

Se utilizó para el estándar de codificación LowerCamelCase que es un tipo de la notación CamelCase.

El estándar de codificación se encuentra en la **Tabla 7-2**, la cual se visualiza a continuación:

**Tabla 7-2:** Estándar de codificación

<b>ESTÁNDAR DE CODIFICACIÓN</b>			
	<b>Tipo</b>	<b>Estándar</b>	<b>Ejemplo</b>
Base de datos	Primary Key	lowerCase	cedula
	Tablas	CamelCase	Cliente
	Atributos	mixedCase	NombreCompleto
Código	Clase	CamelCase	ClienteController
	Método	mixedCase	Total
	Variable	LowerCamelCase	var
	Constante	UPPERCASE	PEU

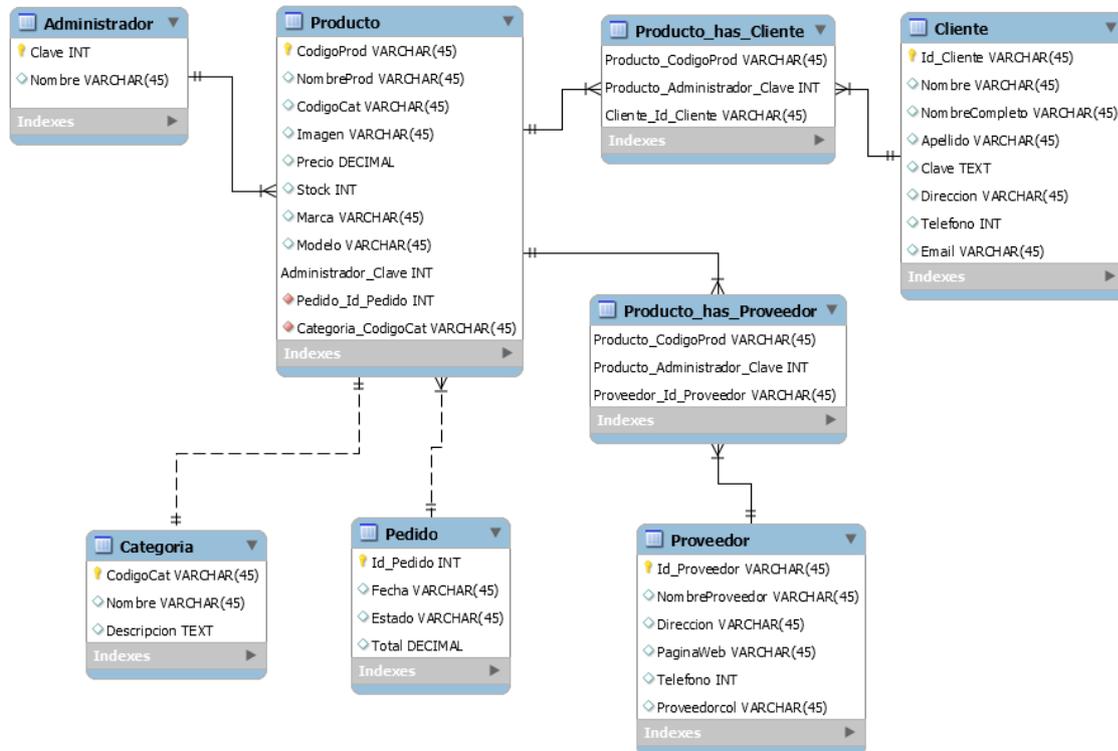
Realizado por: Edwin Hernández, 2019

Una explicación más detallada del estándar de codificación de la aplicación web, se encuentra en el **ANEXO B**.

### 2.4.3. Diseño de la base de datos

Para obtener la base de datos a utilizar se siguió un proceso empezando por un diseño conceptual, así como del modelo entidad-relación y por el proceso de normalización, se obtiene la base de datos final, que está conformada por 8 tablas con un total de 6 claves primarias y 6 claves foráneas.

El diagrama físico de la base de datos de la aplicación web se encuentra en la **Figura 4-2**, que se visualiza a continuación:



**Figura 4-2:** Diagrama del modelo físico de la base de datos  
Realizado por: Edwin Hernández, 2019

#### 2.4.4. Diseño de la interfaz del sistema

El diseño de las interfaces del sistema se realiza para mantener una uniformidad en las plantillas de la aplicación web, el cliente expreso mediante reuniones como quería la interfaz para esto se presentaron diferentes bosquejos de interfaces.

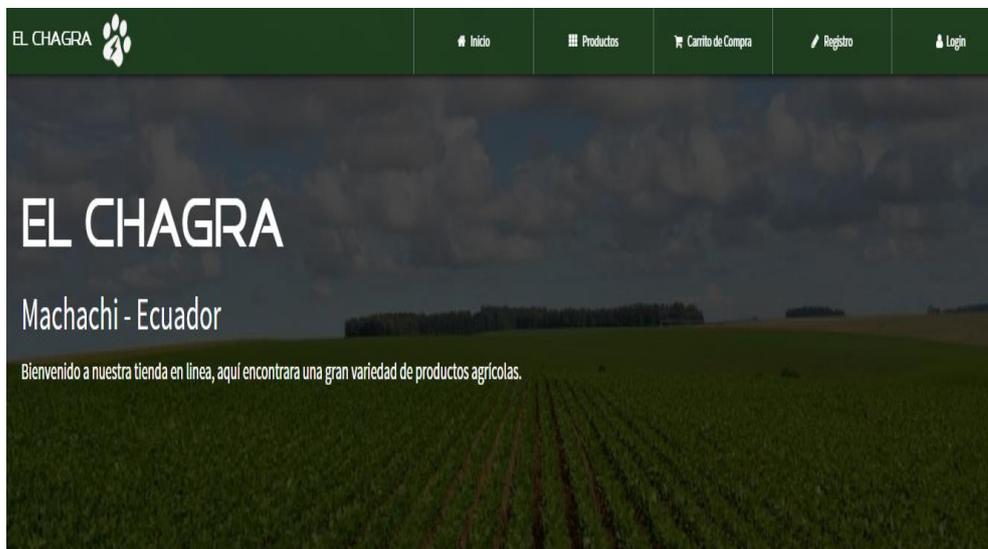
El diseño de las interfaces del sistema se encuentra en el **ANEXO C**.

En la **Figura 5-2**, se muestra el bosquejo de una de las pantallas del sistema con los diferentes elementos que la conforman.



**Figura 5-2:** Bosquejo de gestión de usuarios, productos, proveedores  
 Realizado por: Edwin Hernández, 2019

Luego de presentar los bosquejos de pantallas al cliente, se puede visualizar la interfaz final de la aplicación web en la **Figura 6-2**, quedando conforme con las interfaces de la aplicación web.



**Figura 6-2:** Página de Inicio  
 Realizado por: Edwin Hernández, 2019

En la **Figura 7-2**, se puede observar la interfaz del administrador donde visualiza la información de productos, proveedores, categorías, administrador y pedido con su respectiva gestión de añadir, modificar, eliminar, listar respectivamente.

Administración EMPRESA "EL CHAGRA"

Productos Proveedores Categorías Admin Pedidos

+ Agregar Nuevo

Producto

Mostrar 10 registros Buscar:

Código	Nombre	Categoría	Precio	Descripción del Producto	Unidades	Proveedor	Opciones
001	Romanesco	Semillas	2.00	Destaca por la sanidad de su hoja hasta el momento de la recolección	60	Maria Berrones	 
002	Aminoquelant	Bioestimulantes y Nutrición	3.00	Solución innovadora para tratar las carencias de calcio	60	Vicente Aldaz	 
003	Armurox	Bioestimulantes y Nutrición	5.00	Actúa frente estrés mecánico	80	Johanna Valverde	 
004	CalciPhite	Bioestimulantes y Nutrición	6.00	Mejora la calidad de las plantas	20	Herlinda Argos	 

**Figura 7-2:** Panel del administrador  
Realizado por: Edwin Hernández, 2019

2.4.5. Documentación

Se documenta cada una de las historias de usuario y las historias técnicas evidenciando la realización y funcionamiento de las mismas.

La documentación de una historia de usuario **Tabla 8-2**, para el correcto funcionamiento de la aplicación web; se muestra a continuación:

**Tabla 8-2:** HU-01 Ingresar información de producto

HISTORIA USUARIO	
Número: HU-01	Nombre de la Historia: Ingresar información de producto
Usuario: Desarrollador	Sprint Asignado: 3
Puntos Estimados: 20	Puntos Reales: 20
Descripción: Como desarrollador quiero poder ingresar la información de los productos para conocer la mercadería que posee la empresa.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> <li>Verificar que se haya ingresado la información de los productos en la base de datos.</li> </ul>	

Realizado por: Edwin Hernández, 2019

**Tabla 9-2:** Prueba de Aceptación, HU-01 Ingresar información de producto

PRUEBA DE ACEPTACIÓN	
Código: HU-01.PA-01	Nombre de la Historia: Ingresar información de producto
Nombre de la Prueba: Verificar que se ingrese la información de los productos en la base de datos.	
Responsable: Edwin Hernández	Fecha: 05/02/2019
Descripción: Poder ingresar la información de los productos para conocer los que existen en la empresa.	
Condiciones de Ejecución:	

<ul style="list-style-type: none"> <li>• Estar registrado previamente para poder ingresar la información.</li> <li>•</li> </ul>
Pasos de Ejecución: <ul style="list-style-type: none"> <li>• Dar clic en productos</li> <li>• Dar clic en la opción ingresar</li> <li>• Llenar los datos de los productos</li> <li>• Dar clic en el botón Guardar.</li> </ul>
Resultado Esperado: Se almacenaron los datos en la base de datos del ingreso de productos.
Evaluación de la Prueba: Exitosa

Realizado por: Edwin Hernández, 2019

**Tabla 10-2:** Tarea de Ingeniería, HU-01 Ingresar información de producto

TAREA DE INGENIERIA	
Sprint: 3	Número de tarea: 01
Nombre de la Historia: Ingresar información de producto	
Nombre de la Tarea: Acceso a datos ingresar productos	
Responsable: Edwin Hernández	Tipo de Tarea: Desarrollo
Fecha de Inicio: 01/02/2019	Fecha de Fin: 04/02/2019
Descripción: Crear el método de ingresar producto de la empresa.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> <li>• Verificar que el método ingresar producto este creado.</li> </ul>	

Realizado por: Edwin Hernández, 2019

Se tuvo 28 historias de usuario y 06 historias técnicas, obteniendo 76 tareas de ingeniería y 120 pruebas de aceptación siendo todas exitosas

Las demás historias de usuario y técnicas realizadas para el desarrollo de la aplicación web, se encuentran en el **ANEXO D**.

#### 2.4.6. Manual de usuario

El manual de usuario es el instructivo para el Product Owner que mediante imágenes y los pasos detallados de cada una de las funcionalidades de la aplicación se puede utilizar todas sus funcionalidades e incluye el modo de acceso, ingreso, actualización, consultas, reportes y eliminar la información de acuerdo a los que se quiera realizar.

El manual se encuentra en el **ANEXO E**.

#### 2.5. Fase de cierre

Las actividades realizadas en la terminación de la aplicación web con su respectiva descripción y el responsable, se visualiza a continuación en la **Tabla 11-2:**

**Tabla 11-2:** Actividades de la conclusión de la aplicación web

ACTIVIDAD	DESCRIPCIÓN	RESPONSABLE
Documentar el trabajo de titulación realizado.	Elaboración del manual de usuario y el manual técnico.	Desarrollador
Capacitar al usuario	Entrega del manual de usuario y el manual técnico al Product Owner	Desarrollador

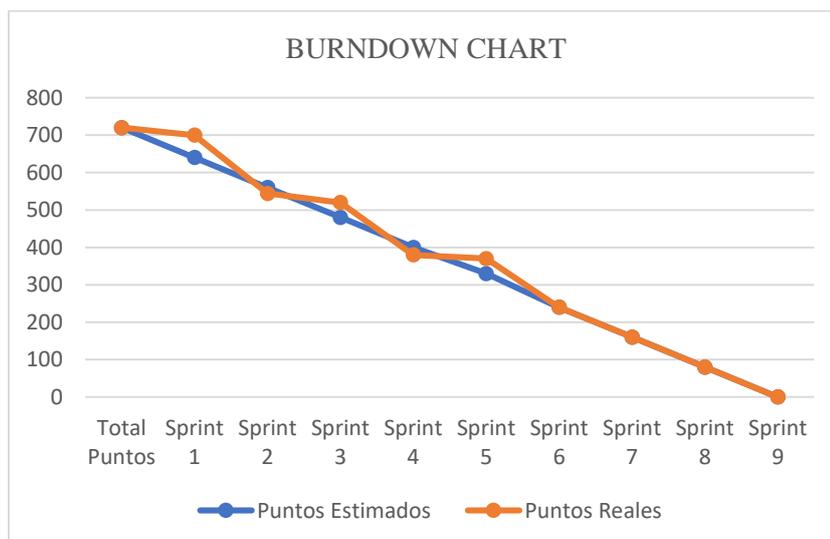
Realizado por: Edwin Hernández, 2019

## 2.6. Gestión del proyecto

### 2.6.1. Burndown Chart

La culminación de los Sprint determina la finalización del desarrollo de la aplicación web; posteriormente se utiliza la herramienta Burndown Chart para la gestión del proyecto **Gráfico 1-2**, donde se puede visualizar el avance del proyecto desde el inicio hasta la finalización del mismo para cumplir con lo planificado.

Se visualiza dos líneas, una de color azul representa lo que se estimó y planifico, mientras que la línea naranja demuestra como fue el avance real del proyecto, las líneas no son iguales como se evidencia por la que en algunos Sprint tomo más tiempo en el desarrollo de lo estimado.



**Gráfico 1-2:** Burndown Chart

Realizado por: Edwin Hernández, 2019.

En el **Gráfico 1-2**, se observa dos líneas, una es de color azul que refleja el camino de los puntos estimados y planificados, mientras que la línea de color naranja muestra el desarrollo del proyecto en realidad, el cual presenta una variación en ambas líneas dado que en el sprint 1 y 3 se comenzó a utilizar las herramientas que se van a utilizar para el desarrollo web, en el sprint 5 se realizaron los ingresos de cada proceso de la aplicación web se ha tomado más tiempo de lo estimado; pero el desarrollo de la aplicación web no se vio afectada porque se trabajó horas extras para que no haya ningún inconveniente en el cumplimiento de los sprint.

## **2.7. Método utilizado para la evaluación de la eficiencia**

Para la evaluación de la eficiencia de los tiempos de respuesta de los procesos de la Empresa “El Chagra”, se lo realizará mediante un estudio comparativo entre la forma automatizada y no automatizada, para lo cual se debe definir primero la población y la muestra

### **2.7.1. Población**

Para evaluar el parámetro de la eficiencia del proceso de pedidos de la empresa “El Chagra” tanto del sistema informático y sin este, la población conforma todos los tiempos de respuesta utilizados para llevar a cabo las diferentes transacciones en el sistema, por lo tanto, la población para el presente estudio es de tipo infinita.

### **2.7.2. Muestra**

Para la obtener el número de datos necesarios para llevar a cabo el análisis estadístico de la eficiencia de ambos sistemas se utilizó la fórmula del tamaño de la muestra para poblaciones infinitas (Aguilar, 2005, p.5), de los procesos más importantes dentro del sistema, como son: registrar cliente, registrar producto, registrar proveedor y registrar pedido.

A continuación, se detalla la fórmula utilizada:

$$n = \frac{Z\alpha^2 * p * q}{d^2}$$

#### **Datos:**

- $Z$ = nivel de confianza (dado a que la seguridad de  $Z\alpha$  es 95% el coeficiente es 1.96).

- p= probabilidad de éxito, o proporción esperada.
- q= probabilidad de fracaso.
- d= precisión (error máximo admisible en términos de proporción)

Con un nivel de confianza del 95% y un error admisible del 6%. Mediante el reemplazo de los datos se tiene el siguiente valor de la muestra:

$$p = 0.5$$

$$q = 1-p = 0.5$$

$$d = 0.05$$

$$n = \frac{(1.96)^2 * 0.5 * 0.5}{(0.06)^2} = 266,77 \text{ valores de tiempo}$$

Se realizó el cálculo de la muestra obteniendo un resultado de 266.77 el mismo que fue redondeado a 267 valores de tiempo.

### **2.7.3. Obtención de datos**

Para la obtención de los datos se utiliza la técnica de observación y el uso de un cronómetro los procesos analizados son registrar cliente, producto, proveedor y pedido; para la toma de tiempos de los procesos no automatizados el gerente inicia llenando de forma manual la ficha de los procesos analizados, mientras que para los tiempos de los procesos automatizados se los realiza mediante la utilización de la aplicación web.

## CAPÍTULO III

### 3. MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS DE RESULTADOS

En el presente capítulo se realizará la evaluación de la aplicación web desarrollada para la Empresa “El Chagra” utilizando la norma ISO/IEC 9126 mediante el análisis de la eficiencia teniendo en cuenta los tiempos de respuesta de la aplicación de los procesos analizados, cumpliendo de esta manera con el objetivo planteado en el trabajo de titulación.

Los tiempos de respuesta fueron tomados en segundos, a partir del lunes 01/06/2019 hasta el viernes 05/06/2019 en horarios de 09:30 a 12:30 y de 14:00 a 17:00. Para la obtención de los tiempos de respuesta de los procesos analizados de forma automatizada se utilizó una laptop ACER, procesador Intel Core i7, con memoria RAM de 8.0 GB y 1 tera de disco duro.

#### 3.1. Análisis de los procesos en la Actualidad de la Empresa

##### 3.1.1. Análisis de Normalidad

El análisis de la normalidad se lo realizó mediante el software estadístico RStudio para los tiempos de respuesta obtenidos en los procesos analizados, **Anexo I**.

- **Planteamiento de la Hipótesis**

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

Hipótesis Alternativa ( $H_1$ ) = Los datos no proviene de una distribución normal.

- **Nivel de significancia**

Se trabajó con un nivel de significancia del  $\alpha=0,06$  ya que es considerado un margen de error de acuerdo a la procedencia de los datos.

- **Estadístico de Prueba**

Los tiempos de respuesta obtenidos de los registros de clientes, productos, proveedores y pedidos de forma no automatizada **Figura 1-3**, que mediante el software RStudio se analizó la

normalidad es estos. Con la utilización de un vector con el nombre de Registrosmanuales, se aplicó Shapiro-Wilk normality test, se obtuvo como resultado el estadístico de prueba ( $w=0.87194$ ) y el valor de  $p$  ( $p\text{-value}=0.1077$ ).

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
manuales.R automatizados.R Untitled1*
Source on Save Run Source
1 Registrosmanuales<-c(
2   200,205,199,204,216,201,204,192,200,211,234,212,221,
3   223,218,238,209,237,215,197,199,239,191,221,196,215,
4   201,229,192,200,211,234,212,221,206,223,218,238,209,
5   191,221,196,215,201,229,192,200,211,234,212,221,206,
6   223,218,238,209,191,221,196,215,201,229,192,200,211,
7   234,236,165,128,121,120,164,128,179,163,159,177,196,
8   195,194,193,197,193,197,198,204,191,198,204,191,197,
9   200,193,196,196,195,194,193,197,193,197,198,204,191,
10  198,204,191,197,200,193,196,196,200,193,196,196,195,
11  194,193,197,193,197,198,204,193,197,198,204,191,198,
12  204,191,197,191,197,128,179,163,159,177,196,195,194,
13  193,204,192,200,211,234,212,221,206,223,218,238,197,
14  199,239,191,221,196,197,128,179,163,159,177,196,195,
15  194,193,204,192,200,211,234,163,159,177,196,195,194,
16  193,204,192,200,211,197,128,179,163,159,177,196,195,
17  194,193,204,192,200,177,196,195,194,193,197,177,196,
18  195,194,193,197,193,197,198,159,177,196,195,194,193,
19  204,192,200,211,199,204,216,195,194,193,197,193,197,
20  198,159,177,196,195,194,193,204,192,200,211,199,204,
21  216,159,177,196,195,194,193,204,192,200,194,193,204,
22  192,200,211,199,204,216
23 )
24 shapiro.test(Registrosmanuales)

Console Terminal
> Registrosmanuales<-c(
+ 200,205,199,204,216,201,204,192,200,211,234,212,221,
+ 223,218,238,209,237,215,197,199,239,191,221,196,215,
+ 201,229,192,200,211,234,212,221,206,223,218,238,209,
+ 191,221,196,215,201,229,192,200,211,234,212,221,206,
+ 223,218,238,209,191,221,196,215,201,229,192,200,211,
+ 234,236,165,128,121,120,164,128,179,163,159,177,196,
+ 195,194,193,197,193,197,198,204,191,198,204,191,197,
+ 200,193,196,196,195,194,193,197,193,197,198,204,191,
+ 198,204,191,197,200,193,196,196,200,193,196,196,195,
+ 194,193,197,193,197,198,204,193,197,198,204,191,198,
+ 204,191,197,191,197,128,179,163,159,177,196,195,194,
+ 193,204,192,200,211,234,212,221,206,223,218,238,197,
+ 199,239,191,221,196,197,128,179,163,159,177,196,195,
+ 194,193,204,192,200,211,234,163,159,177,196,195,194,
+ 193,204,192,200,211,197,128,179,163,159,177,196,195,
+ 194,193,204,192,200,177,196,195,194,193,197,177,196,
+ 195,194,193,197,193,197,198,159,177,196,195,194,193,
+ 204,192,200,211,199,204,216,195,194,193,197,193,197,
+ 198,159,177,196,195,194,193,204,192,200,211,199,204,
+ 216,159,177,196,195,194,193,204,192,200,194,193,204,
+ 192,200,211,199,204,216
+ )
> shapiro.test(Registrosmanuales)

      shapiro-wilk normality test

data:  Registrosmanuales
w = 0.87194, p-value = 0.1077

```

**Figura 1-3:** Tiempos manuales de los registros  
Realizado por: Edwin Hernández, 2019

Los tiempos de respuesta obtenidos de los registros de clientes, productos, proveedores y pedidos de forma automatizada **Figura 2-3**, que mediante el software RStudio se analizó la normalidad es estos. Con la utilización de un vector con el nombre de Registrosautomatizados, se aplicó Shapiro-Wilk normality test, se obtuvo como resultado el estadístico de prueba ( $w=0.74718$ ) y el valor de  $p$  ( $p\text{-value}=0.179$ ).

```

1 Registrosautomatizados<-c(
2 45,48,47,45,53,57,50,51,40,42,48,51,50,52,58,123,123,
3 121,123,122,120,121,120,124,124,125,124,121,122,122,
4 120,121,120,124,124,125,124,121,122,122,120,121,120,
5 124,124,125,124,121,122,122,120,121,120,124,124,125,
6 124,121,122,122,120,121,120,124,124,125,124,124,82,
7 95,91,69,76,104,96,62,89,79,66,65,66,67,65,64,66,64,
8 63,66,63,64,67,68,65,63,66,67,63,66,63,64,67,68,65,
9 63,66,67,63,66,63,64,67,68,65,63,66,67,63,66,63,64,
10 67,68,65,63,66,67,63,66,63,64,67,68,65,63,65,67,123,
11 123,121,123,122,120,121,120,124,124,125,124,121,122,
12 66,65,66,67,65,64,66,67,68,65,63,66,67,67,65,64,66,
13 67,68,65,63,66,67,67,65,64,66,67,68,65,63,66,65,67,
14 65,64,66,67,68,65,63,66,67,67,65,64,66,67,68,65,63,
15 69,76,104,96,62,89,79,66,65,66,67,65,124,124,125,96,
16 62,89,79,66,65,66,67,65,64,89,79,66,65,66,65,66,67,
17 65,64,89,79,66,65,66,65,66,67,65,64,89,79,66,66,66,
18 65,66,67,65,64,89,79,66,65,66,65,66,67,65,64,89,79
19 )
20 shapiro.test(Registrosautomatizados)

```

```

> Registrosautomatizados<-c(
+ 45,48,47,45,53,57,50,51,40,42,48,51,50,52,58,123,123,
+ 121,123,122,120,121,120,124,124,125,124,121,122,122,
+ 120,121,120,124,124,125,124,121,122,122,120,121,120,
+ 124,124,125,124,121,122,122,120,121,120,124,124,125,
+ 124,121,122,122,120,121,120,124,124,125,124,124,82,
+ 95,91,69,76,104,96,62,89,79,66,65,66,67,65,64,66,64,
+ 63,66,63,64,67,68,65,63,66,67,63,66,63,64,67,68,65,
+ 63,66,67,63,66,63,64,67,68,65,63,66,67,63,66,63,64,
+ 67,68,65,63,66,67,63,66,63,64,67,68,65,63,65,67,123,
+ 123,121,123,122,120,121,120,124,124,125,124,121,122,
+ 66,65,66,67,65,64,66,67,68,65,63,66,67,67,65,64,66,
+ 67,68,65,63,66,67,67,65,64,66,67,68,65,63,66,65,67,
+ 65,64,66,67,68,65,63,66,67,67,65,64,66,67,68,65,63,
+ 69,76,104,96,62,89,79,66,65,66,67,65,124,124,125,96,
+ 62,89,79,66,65,66,67,65,64,89,79,66,65,66,65,66,67,
+ 65,64,89,79,66,65,66,65,66,67,65,64,89,79,66,66,66,
+ 65,66,67,65,64,89,79,66,65,66,65,66,67,65,64,89,79
+ )
> shapiro.test(Registrosautomatizados)

      shapiro-wilk normality test

data:  Registrosautomatizados
W = 0.74718, p-value = 0.179

```

**Figura 2-3:** Tiempos automatizados de los registros  
**Realizado por:** Edwin Hernández, 2019

El mismo procedimiento del análisis de normalidad de datos se realizó para los procesos de registrar cliente, registrar producto, registrar proveedor y registrar pedido los cuales se encuentran detallados en el ANEXO G.

- **Regla de decisión**

Mediante la obteniendo de los valores anteriores se continua con la toma de decisión para lo cual se utilizó el valor de p, para la regla siendo esta  $p > \alpha$ , donde si cumple esta condición No se rechaza  $H_0$ , es decir los datos provienen de una distribución normal.

En la **Tabla 1-3**, se muestran los resultados para la toma de decisión de los tiempos de los procesos de forma no automatizada y automatizada:

**Tabla 1-3:** Resultados para la toma de decisión

<b>TIEMPOS DE LOS PROCESOS NO AUTOMATIZADOS (SEG)</b>	<b>TIEMPOS DE LOS PROCESOS AUTOMATIZADOS (SEG)</b>
<b>Registros</b>	
$p > \alpha$ No rechazo $H_0$ 0.1077 > 0.06	$p > \alpha$ No rechazo $H_0$ 0.179 > 0.06

Realizado por: Edwin Hernández, 2019

- **Toma de decisión**

Dado que el valor de  $p$  en los procesos analizados es mayor que el nivel de significancia de 0.06, no se rechaza  $H_0$ , es decir que los datos provienen de una distribución normal.

Los resultados de la estadística descriptiva e inferencial de los procesos registrar cliente, registrar producto, registrar proveedor y registrar pedido se encuentran en el **ANEXO H**.

### 3.1.2. Estadística Descriptiva

Mediante la estadística descriptiva se realiza el análisis de una manera general de los tiempos que fueron recopilados y posteriormente su respectiva comparación.

En los procesos analizados los tiempos se expresaron en segundos, permitiendo evaluar la eficiencia del sistema, para lo cual se obtuvieron la media, desviación estándar, valor mínimo y máximo de los tiempos.

En la **Tabla 2-3**, se visualiza los tiempos obtenidos de cada uno de los procesos de acuerdo al método no automatizado y el método automatizado.

**Tabla 2-3:** Estadística descriptiva de los registros

<b>TIEMPO EN SEGUNDOS DEL MÉTODO NO AUTOMATIZADO</b>		<b>TIEMPO EN SEGUNDOS DEL MÉTODO AUTOMATIZADO</b>	
<b>Media</b>	196.5964	<b>Media</b>	82.9385
<b>Desviación estándar</b>	11.32	<b>Desviación estándar</b>	7.414
<b>Mínimo</b>	120	<b>Mínimo</b>	40
<b>Máximo</b>	239	<b>Máximo</b>	125

Realizado por: Edwin Hernández, 2019

### 3.1.3. Estadística Inferencial

Para el análisis de los datos se utilizó la herramienta de Análisis de Datos de Microsoft Excel y el software SIAE 2.0.

- **Planteamiento de la Hipótesis**

Hipótesis Nula ( $H_0$ ) = El tiempo promedio automatizado es igual al tiempo promedio no automatizado para cada registro.

Hipótesis Alternativa ( $H_1$ ) = El tiempo promedio automatizado es diferente al tiempo promedio no automatizado para cada registro.

- **Nivel de significancia**

Se utilizó un nivel de significancia del  $\alpha=0.06$  ya que es considerada un margen de error mínimo.

- **Estadístico de Prueba**

Se aplicó la inferencia estadística mediante la utilización de una prueba paramétrica, dado que el número de datos de las muestras en los grupos de estudio es mayor a 30, se utilizó la prueba Z para dos muestras independientes. Los datos obtenidos del cálculo de la prueba estadística se encuentran en la **Tabla 3-3**.

**Tabla 3-3:** Distribución Z con relación a los registros

<b>PRUEBA T PARA MEDIAS DE DOS MUESTRAS EMPAREJADAS</b>	
Nivel de significancia	6%
Nivel de confianza	95%
Valor critico	1.9
Z	33.96

Realizado por: Edwin Hernández, 2019

- **Regla de decisión**

En el **Gráfico 3-3** se muestra la representación gráfica de la distribución normal para medias de dos muestras emparejadas, donde se trabajó con un nivel de significancia del 6% y un nivel de confianza del 95%, un valor crítico de 1,9 valor obtenido en base a las tablas de distribución normal estándar; y el valor de prueba 33,96 determinado en base a la aplicación de la prueba de Z.

A continuación, en el **Gráfico 1-3** se muestra la gráfica obtenida:

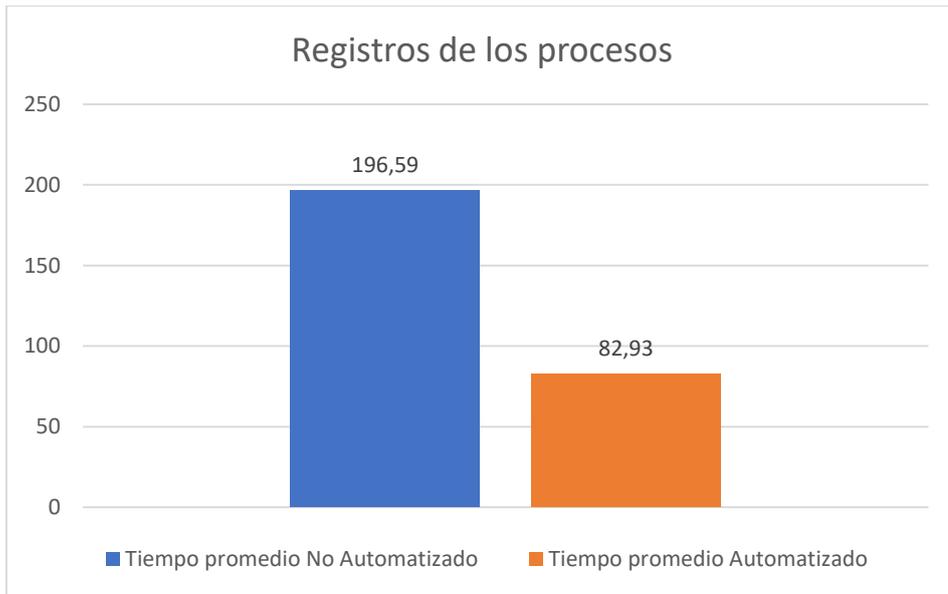


**Gráfico 1-3:** Gráfica de Distribución Normal  
Realizado por: Edwin Hernández, 2019

- **Toma de decisión**

El valor de Z cae fuera de la región de aceptación; por lo cual, con un nivel de significancia del 6%, se rechaza la hipótesis nula y se acepta la alternativa. Es decir que el tiempo promedio automatizado es estadísticamente diferente al tiempo promedio no automatizado para los registros analizados.

En el **Gráfico 2-3**, se visualiza la diferencia del tiempo de respuesta de los registros tanto de forma automatizada y no automatizada.



**Gráfico 2-3:** Tiempo promedio de los registros  
**Realizado por:** Edwin Hernández, 2019

Los tiempos disminuyeron y para su representación se utiliza las medias de los promedios con y sin la aplicación web los cuales son 196.59 y 82.93 segundos respectivamente **Gráfico 2-3**, notándose una reducción de 57,81% en los tiempos de ejecución de los procesos mediante la utilización de la aplicación web.

## CONCLUSIONES

- Mediante el análisis de los procesos en la empresa “El Chagra” se determinó que intervienen cuatro actores principales: el cliente, empleado, administrador y proveedor; se evidencia así también la existencia tres procesos registro de cliente (3 actividades), gestión de pedido (13 actividades) y entrega del pedido (1 actividad) teniendo un total de 17 actividades. Aproximadamente la empresa lleva a cabo 65 procesos al día en promedio.
- Para la estructuración y la separación en capas de la aplicación web aplicando el patrón Modelo Vista Controlador (MVC) se utilizó en la capa de modelo 7 clases, para la capa vista 8 páginas y en la capa controlador 9 clases.
- Con la utilización de la metodología Scrum, durante el proceso de desarrollo de software se tuvo una duración de 720 horas obteniendo 76 tareas de ingeniería, 120 pruebas de aceptación con un total de 28 historias de usuario y 06 historias técnicas.
- Para el uso del framework Bootstrap y el pre-compilador Sass se creó carpetas css (6 hojas de estilo) y javascript (9 archivos de scripts). Se utilizaron un total de 24 componentes del framework entre los cuales destacan grid, cards, fronts, navbar, form, modal.
- Es importante también destacar que se evidenció un mejoramiento en la eficiencia en los indicadores de los procesos analizados, el cual se llevó a cabo utilizando la prueba estadística t- Student, obteniendo los siguientes porcentajes de mejora: Para el registro de clientes un 76%, registro de productos un 43%, registro de proveedores un 43%, registro de pedidos un 66% con un nivel de significancia de 0,03.
- Consolidando todos los datos de las pruebas realizadas, con un total de 119 datos, con un error del 0.03 y un nivel de confianza de un 95% se evidenció mediante la prueba estadística inferencial Z que existe una diferencia significativa de 1.96 a favor del proceso con el sistema informático con lo cual aplicando la fórmula de la variación porcentual arrojó como resultado 57,81%.

## RECOMENDACIONES

- La aplicación desarrollada contempla el proceso de gestión de pedidos y puede ser escalable, por lo que se recomienda ampliar a futuro e incluir el módulo de pagos utilizando PayPal.
- Es recomendable que se realice una capacitación previa de las tecnologías a utilizar para evitar desfases en el tiempo del proyecto.
- El framework Bootstrap con el pre-compilador Sass permitió mejorar la eficiencia del sistema, pero se propone otro tipo de análisis como la usabilidad, productividad, seguridad, etc.
- Evaluar la eficiencia de las aplicaciones web aplicando la norma ISO/IEC 25000 por ser actualizada y que reemplazó a la ISO/IEC 9126.

## BIBLIOGRAFÍA

**ACEDO, José.** *Web: ¿Qué es el Framework Bootstrap? Ventajas y Desventajas. – Apuntes de Programación.* [Blog]. [Consulta: 8 julio 2019]. Disponible en: <http://programacion.jias.es/2015/05/web-¿que-es-el-framework-bootstrap-ventajas-desventajas/>.

**AGUILAR, S.** *Salud en Tabasco.* [en línea], pp. 333-338. [Consulta: 28 noviembre 2019]. ISSN 1405-2091. Disponible en: <http://www.redalyc.org/articulo.oa?id=48711206>.

**ALMAGRO, C.** *Lenguajes de Programación Capítulo 1. Introducción.* [en línea]. [Consulta: 9 julio 2019]. Disponible en: <http://java.sun.com/docs/books/jls/>.

**ALVAREZ, I.** *Desarrollo Ágil con Desarrollo Ágil con SCRUM.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:sg07.p02.scrum.pdf>.

**ALVAREZ, M.** *Materialize CSS.* [en línea]. [Consulta: 8 julio 2019]. Disponible en: <https://desarrolloweb.com/articulos/materialize-framework-css.html>.

**ARCE, A.** *Programación PHP.* [en línea]. [Consulta: 9 julio 2019]. Disponible en: <https://buildmedia.readthedocs.org/media/pdf/programacion-php/latest/programacion-php.pdf>.

**ARCOS, G. MENÉNDEZ, J. & VALLEJO, J.** *Comparative Study of Performance and Productivity of MVC and MVVM design patterns.* *KnE Engineering* [en línea], vol. 1, no. 2, pp. 241. [Consulta: 4 julio 2019]. ISSN 2518-6841 Disponible en: <https://knepublishing.com/index.php/KnE-Engineering/article/view/1498>.

**BARBER, F. & FERRÍS, R.** *Lenguajes de programación.* [en línea]. [Consulta: 1 abril 2019]. Disponible en: [http://informatica.uv.es/iiguia/AED/oldwww/2004\\_05/AED.Tema.02.pdf](http://informatica.uv.es/iiguia/AED/oldwww/2004_05/AED.Tema.02.pdf).

**BASCÓN, E.** *El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en*

*Java Swing*. [en línea]. [Consulta: 4 julio 2019]. Disponible en: <https://pdfs.semanticscholar.org/4a4a/e13ad1320ee1baf884ef6a91b5010c64a1a4.pdf>.

**BAUTISTA, Eliseo.** *Lenguajes de programación: RUBY*. [Blog]. [Consulta: 9 julio 2019]. Disponible en: <http://aplicaciones-web-lenguajes-programaci.blogspot.com/2011/12/ruby.html>.

**BERMEO, Alejandro.** Estudio comparativo de los patrones para interfaces de usuario mvvm y mvc. aplicado al desarrollo del sitio de gestión de ventas para vidrialum. [en línea]. (Tesis Pregrado). Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería en Sistemas. Riobamba, Ecuador. [Consulta: 4 julio 2019]. Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/3559/1/18T00567.pdf>.

**BORBÓN, Nuvia.** *Norma de evaluación iso/iec 9126 | evaluacion de software*. [Blog]. [Consulta: 5 julio 2019]. Disponible en: <http://actividadreconocimiento-301569-8.blogspot.com/2013/03/norma-de-evaluacion-isoiec-9126.html>.

**BLANCH, Alberto.** *Si trabajas con CSS, tienes que conocer el preprocesador Sass* [Blog]. [Consulta: 9 julio 2019]. Disponible en: <https://www.arsys.es/blog/programacion/que-es-sass/>.

**BLASCO-LERMA, R.** *Aplicaciones web* [en línea]. [Consulta: 2 julio 2019]. Disponible en: <https://ebookcentral.proquest.com/lib/esPOCHsp/reader.action?docID=3213262&query=aplicaciones%2Bweb>.

**CAMPAÑA, R.** *Ingeniería del software : El proceso de desarrollo rápido de aplicaciones (DRA) de software: Un aporte práctico en el Instituto Geográfico Militar*. [en línea]. [Consulta: 24 junio 2019]. ISBN 9701054733. Disponible en: [https://www.researchgate.net/publication/303839299\\_El\\_proceso\\_de\\_desarrollo\\_rapido\\_de\\_aplicaciones\\_DRA\\_de\\_software\\_Un\\_aporte\\_practico\\_en\\_el\\_Instituto\\_Geografico\\_Militar](https://www.researchgate.net/publication/303839299_El_proceso_de_desarrollo_rapido_de_aplicaciones_DRA_de_software_Un_aporte_practico_en_el_Instituto_Geografico_Militar).

**CAMPO, G.** *Patrones de Diseño, Refactorización y Antipatrones. Ventajas y Desventajas de su Utilización en el Software Orientado a Objetos*. [en línea]. [Consulta: 4 julio 2019]. Disponible en: <https://www.ucasal.edu.ar/htm/ingenieria/cuadernos/archivos/4-p101-Campo.pdf>.

**CELAYA, A.** *Programación*. [en línea]. [Consulta: 18 mayo 2019]. Disponible en: <http://www.iqcelaya.itc.mx/~vicente/Programacion/Lenguajes.pdf>.

**CHAMBI, A.** *Manual SASS* [en línea]. [Consulta: 9 julio 2019]. Disponible en: [https://www.academia.edu/35937550/SASS\\_Manual\\_Javier\\_Guede](https://www.academia.edu/35937550/SASS_Manual_Javier_Guede).

**COBO, Á. GÓMEZ, P. PÉREZ, D. & ROCHA, R.** *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web*. [en línea]. [Consulta: 9 julio 2019]. Disponible en: [www.diazdesantos.es/ediciones](http://www.diazdesantos.es/ediciones).

**CRISTIÁ, M.** *Catálogo Incompleto de Estilos Arquitectónicos*. [en línea]. [Consulta: 4 julio 2019]. Disponible en: <https://pdfs.semanticscholar.org/ac17/7da0f8bb600c1ad09c1851f4f385d30db85f.pdf>.

**CULOCCIONI, S.** *Desarrollo de software patrones de diseño con pHp 5* [en línea]. [Consulta: 16 julio 2019]. Disponible en: <https://www.solvetic.com/tutoriales/article/1487-desarrollo-de-software-patrones-de-diseño-con-php-5/>.

**DEEMER, P.** *Una introducción básica a la teoría y práctica de Scrum Versión 2.0*. [en línea]. [Consulta: 28 mayo 2019]. Disponible en: [www.goodagile.com](http://www.goodagile.com).

**DURÁN, D.** *Gestión de la calidad de productos editoriales multimedia*. [en línea]. [Consulta: 22 julio 2019]. Disponible en: <https://ebookcentral.proquest.com/lib/espochsp/reader.action?docID=5214364&ppg=1>.

**ESPINOZA-MEZA, Arnaldo.** *Manual para elegir una metodología de desarrollo de software dentro de un proyecto informático*. [en línea]. (Tesis Pregrado). Universidad de Piura, Facultad de Ingeniería, Área departamental de Ingeniería Industrial y de Sistemas. Piura, Perú. [Consulta: 5 julio 2019]. Disponible en: [https://pirhua.udep.edu.pe/bitstream/handle/11042/2747/ING\\_521.pdf?sequence=1&isAllowed=y](https://pirhua.udep.edu.pe/bitstream/handle/11042/2747/ING_521.pdf?sequence=1&isAllowed=y).

**FERNÁNDEZ, J.** *Calidad del software*. [en línea]. [Consulta: 24 junio 2019]. Disponible en:

[https://www.uv.mx/personal/jfernandez/files/2010/07/8\\_Calidad.pdf](https://www.uv.mx/personal/jfernandez/files/2010/07/8_Calidad.pdf).

**FIGUEROA, M.** *Calidad en la Industria del Software. La Norma ISO-9126.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <http://www.monografias.com/trabajos5/>.

**FONTELA, C.** *Diseño MVC.* [en línea]. [Consulta: 4 julio 2019]. Disponible en: [http://materias.fi.uba.ar/7507/content/20102/teoricas/7507\\_09\\_diseno\\_mvc.pdf](http://materias.fi.uba.ar/7507/content/20102/teoricas/7507_09_diseno_mvc.pdf).

**GALLEGO, M.** *Metodología Scrum.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>.

**GARCÍA, N.** *Lenguaje de programación Php.* [en línea]. [Consulta: 9 julio 2019]. Disponible en: [https://nellygarcia.files.wordpress.com/2010/04/clase\\_4\\_5.pdf](https://nellygarcia.files.wordpress.com/2010/04/clase_4_5.pdf).

**GARRIDO, J.** *Arquitectura y diseño de sistemas web modernos.* [en línea], [Consulta: 17 mayo 2019]. Disponible en: [http://pegaso.ls.fi.upm.es/~sortega/html\\_css/files/Arquitectura\\_y\\_diseno\\_de\\_sistemas\\_web\\_modernos.pdf](http://pegaso.ls.fi.upm.es/~sortega/html_css/files/Arquitectura_y_diseno_de_sistemas_web_modernos.pdf).

**GÓMEZ, E.** *Bases de Datos 1* [en línea]. [Consulta: 27 mayo 2019]. Disponible en: <http://www.dlsi.ua.es/asignaturas/bd>.

**GÓMEZ, Rodrigo.** *Modelo Vista Controlador.* [Blog]. [Consulta: 4 julio 2019]. Disponible en: <http://rodrigogr.com/blog/modelo-vista-controlador/>.

**GONZÁLEZ, J.** *Desarrollo de sitios web con PHP y MySQL* [en línea]. [Consulta: 27 mayo 2019]. Disponible en: <http://www.lsi.us.es/cursos/cursophp/apuntes/tema3.pdf>.

**GONZALES, M.** *Arquitectura cliente servidor.* [en línea]. [Consulta: 16 julio 2019]. Disponible en: <https://es.slideshare.net/ester.gonzalez/arquitectura-cliente-servidor-104004600>.

**GUTIERREZ, J.** *Definición arquitectura cliente servidor.* [en línea]. [Consulta: 2 julio 2019].

Disponible en: [www.monografias.com](http://www.monografias.com).

**GLERA, Cristina.** Desarrollo de una guía para dispositivos móviles de establecimientos para celiacos en logroño. [en línea]. (Tesis Pregrado). Universidad Pública de Navarra, Escuela Técnica Superior de Ingenieros Industriales y de Telecomunicación. Navarra, España. [Consulta: 2 julio 2019]. Disponible en: <http://academica-e.unavarra.es/bitstream/handle/2454/7545/578082.pdf?sequence=1>.

**HARO, J. & CAMPS, J.** *Diseño e implementación de un marco de trabajo (framework) de presentación para aplicaciones JEE.* [en línea]. [Consulta: 8 julio 2019]. Disponible en: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/876/1/00765tfc.pdf>.

**HERNÁNDEZ, J.** *Ingeniería del Software II.* [en línea]. [Consulta: 24 junio 2019]. Disponible en: <https://ocw.unican.es/pluginfile.php/1408/course/section/1803/tema2-alidadSistemasSoftware.pdf>.

**HUAMAN, W.C.** *Los 10 patrones comunes de arquitectura de software.* [en línea]. [Consulta: 4 julio 2019]. Disponible en: <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>.

**IPIÑA, D.** *Introducción a Ruby on Rails Filosofía DRY-COC.* [en línea]. [Consulta: 9 julio 2019]. Disponible en: <http://rubyforge.org/frs/download.php/2407/ruby182-14.exe>.

**KEN, S.** *La Guía de Scrum* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf>.

**LOCKHART, T.** *Manual del usuario de PostgreSQL El equipo de desarrollo de PostgreSQL.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <http://index-of.co.uk/SERVIDORES/Postgres-User.pdf>.

**LÓPEZ, B.** *Metodología Ágil de Desarrollo de Software-XP.* [en línea]. [Consulta: 24 junio 2019]. Disponible en: [http://www.runayupay.org/publicaciones/2244\\_555\\_COD\\_18\\_290814203015.pdf](http://www.runayupay.org/publicaciones/2244_555_COD_18_290814203015.pdf).

**LOZANO, I.** *¿Qué es un preprocesador de CSS?* [en línea]. [Consulta: 8 julio 2019]. Disponible en: <https://abalozz.es/que-es-un-preprocesador-de-css/>.

**MARINI, E.** *Arquitectura Cliente Servidor.* [en línea]. [Consulta: 2 julio 2019]. Disponible en: [www.linuxito.com](http://www.linuxito.com).

**MARIÑO, C.** *Programacion en Php5 Nivel básico.* [en línea]. [Consulta: 9 julio 2019]. Disponible en: [http://administraciondesistemas.pbworks.com/f/Manual\\_PHP5\\_Basico.pdf](http://administraciondesistemas.pbworks.com/f/Manual_PHP5_Basico.pdf).

**MARTÍN, J.** *Metodología SCRUM para desarrollo de software a medida.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>.

**MARTÍNEZ, Carlos.** *Fundamentos y evolución de la multimedia Las aplicaciones web y las bases de datos.* [Blog]. [Consulta: 16 julio 2019]. Disponible en: <http://multimedia.uoc.edu/blogs/fem/es/las-aplicaciones-web-y-las-bases-de-datos/>.

**MAZZA, Marcelo.** *Introducción a PostCSS.* [Blog]. [Consulta: 9 julio 2019]. Disponible en: <https://octuweb.com/introduccion-a-postcss/>.

**MENDOZA, Melissa.** *Ventajas del uso de LESS* [Blog]. [Consulta: 9 julio 2019]. Disponible en: <https://blog.michelletores.mx/ventajas-del-uso-de-less/>.

**MENZINSKY, A.** *Scrum Manager.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <http://www.streetsofdublin.com/>.

**MESTRAS, J.** *Patrones de diseño orientado a objetos Programación Orientada a Objetos.* [en línea]. [Consulta: 15 julio 2019]. Disponible en: <https://www.fdi.ucm.es/profesor/jpavon/poo/2.14PDOO.pdf>.

**MESTRAS, J.** *Estructura de las Aplicaciones Orientadas a Objetos El patrón Modelo-Vista-Controlador (MVC).* [en línea]. [Consulta: 17 mayo 2019]. Disponible en:

<http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html>.

**MIRÓ, Albert.** *7 características del lenguaje PHP que lo convierten en uno de los más potentes.* [en línea]. [Consulta: 9 julio 2019]. Disponible en: <https://www.deustoformacion.com/blog/programacion-diseno-web/7-caracteristicas-lenguaje-php-que-lo-convierten-uno-mas-potentes>.

**MORA, Sergio.** *Programación de aplicaciones web* [Blog]. [Consulta: 2 julio 2019]. Disponible en: <https://gplsi.dlsi.ua.es/~slujan/materiales/pi-cliente2-muestra.pdf>.

**NAVARRETE, E. & NAVARRO, D.** *Desarrollo de un marco de trabajo (framework) para el desarrollo de aplicaciones web en la Universidad Nacional de Costa Rica.* [en línea]. S.I.: [Consulta: 5 julio 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).

**OCAÑA, Jesús.** *Materialize CSS Framework responsive de Google.* [Blog]. [Consulta: 8 julio 2019]. Disponible en: <https://www.azulweb.net/materialize-css-framework-responsive-google/>.

**ORACLE,** *Oracle Database 11g en Windows: Desarrollo e implementación.* [en línea]. [Consulta: 27 mayo 2019]. Disponible en: <https://www.oracle.com/technetwork/es/documentation/317481-esa.pdf>.

**ORFALI, R.** *Cliente/Servidor.* [en línea]. [Consulta: 3 julio 2019]. Disponible en: <http://siul02.si.ehu.es/~jimena/ABD/fuentes/ClienteServidor.pdf>.

**ORTEGA, D.** *¿Qué es PostCSS? Introducción | OpenWebinars.* [en línea]. [Consulta: 9 julio 2019]. Disponible en: <https://openwebinars.net/blog/que-es-postcss-introduccion/>.

**PALACIO, J.** *Scrum Manager y las reglas de scrum.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <http://www.streetsofdublin.com/>.

**PASTOR, J.** *Estudio y clasificación de tipos de aplicaciones web y determinación de atributos de usabilidad más relevantes.* [en línea]. [Consulta: 2 julio 2019]. Disponible en: <https://riunet.upv.es/bitstream/handle/10251/32839/Memoria.pdf>.

**PEÑA, D.** *Tipos de Arquitecturas de Software.* [en línea]. [Consulta: 2 julio 2019]. Disponible en: <https://prezi.com/2uraxobilh0t/tipos-de-arquitecturas-de-software/>.

**PERALTA, A.** *Metodología SCRUM.* [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <https://fi.ort.edu.uy/innovaportal/file/2021/1/scrum.pdf>.

**PÉREZ, J.** *Css avanzado.* [en línea]. [Consulta: 8 julio 2019]. Disponible en: [www.librosweb.es/avanzado](http://www.librosweb.es/avanzado).

**PÉREZ, C.** *Bases de datos.* [en línea]. [Consulta: 27 mayo 2019]. Disponible en: [www.glo.org.mx](http://www.glo.org.mx).

**RAMÍREZ, I.** *Manual oficial de SASS.* [en línea]. [Consulta: 8 julio 2019]. Disponible en: [https://webkode.es/wp-content/uploads/docs/css/7\\_Manual\\_de\\_sass.pdf](https://webkode.es/wp-content/uploads/docs/css/7_Manual_de_sass.pdf).

**RIVERA, A.** *Estudio de la arquitectura de software.* [en línea]. [Consulta: 4 julio 2019]. Disponible en: [http://repositorio.utn.edu.ec/bitstream/123456789/988/2/04\\_ISC\\_164\\_Informe\\_Tecnico.pdf](http://repositorio.utn.edu.ec/bitstream/123456789/988/2/04_ISC_164_Informe_Tecnico.pdf).

**RODRÍGUEZ, T.** *Bootstrap, framework de twitter.* [en línea]. [Consulta: 8 julio 2019]. Disponible en: <https://www.genbeta.com/desarrollo/bootstrap>.

**RUIZ, E.** *Metodología de desarrollo de software.* [en línea]. [Consulta: 5 julio 2019]. Disponible en: [www.uladech.edu.pe](http://www.uladech.edu.pe).

**RUIZ, S.** *Arquitectura Monolítica y Arquitectura Cliente - Servidor.* [en línea]. [Consulta: 2 julio 2019]. Disponible en: [https://sandramarramirez.fandom.com/es/wiki/Arquitectura\\_Monolítica\\_y\\_Arquitectura\\_Cliente](https://sandramarramirez.fandom.com/es/wiki/Arquitectura_Monolítica_y_Arquitectura_Cliente)

\_ \_Servidor.

**SALINAS, J.** *El Lenguaje de Programación Java*. [en línea]. [Consulta: 9 julio 2019]. Disponible en: <https://www.inf.utfsm.cl/~mcloud/iwi-253/tareas/t2-lp-2004-Ayudantia1.pdf>.

**SÁNCHEZ, J.** *MySQL guía rápida (versión Windows)*. [en línea]. [Consulta: 27 mayo 2019]. Disponible en: <http://www.mysql.com>.

**SARRÍA, F.** *Programación en SQL con PostgreSQL*. [en línea]. [Consulta: 27 mayo 2019]. Disponible en: <https://www.um.es/geograf/sigmur/sigpdf/postgresql.pdf>.

**SHELDON, R.** *Fundamentos de SQL* [en línea]. [Consulta: 27 mayo 2019]. Disponible en: [www.detodoprogramacion.com](http://www.detodoprogramacion.com).

**SCHIAFFARINO, A.** *Modelo cliente servidor: ¿Qué es? Características, Ventajas y Desventajas*. [en línea]. [Consulta: 3 julio 2019]. Disponible en: <https://blog.infranetworking.com/modelo-cliente-servidor/>.

**SEGOVIA, E.** *Tema 2: El modelo cliente/servidor*. [en línea]. [Consulta: 17 mayo 2019]. Disponible en: [https://www.infor.uva.es/~fdiaz/sd/2005\\_06/doc/SD\\_TE02\\_20060305.pdf](https://www.infor.uva.es/~fdiaz/sd/2005_06/doc/SD_TE02_20060305.pdf).

**SERRANO, D. & IZQUIERDO, J.** *Tutorial de Prácticas del gestor de Bases de Datos MySql*. [en línea]. (Tesis pregrado). Universidad del Azuay, Facultad de Ciencias de la Administración, Escuela de Ingeniería de Sistemas. Cuenca, Ecuador. [Consulta: 28 mayo 2019]. Disponible en: <http://dspace.uazuay.edu.ec/bitstream/datos/2326/1/06827.pdf>.

**SOLIS, J.** *¿Qué es Bootstrap y cómo funciona en el diseño web?*. [en línea]. [Consulta: 8 julio 2019]. Disponible en: <https://www.arweb.com/chucherias/¿que-es-bootstrap-y-como-funciona-en-el-diseno-web/>.

**TERALCO, A.** *Scrum, un caso práctico*. [en línea]. [Consulta: 28 mayo 2019]. Disponible en: <http://www.dtic.ua.es/jdare10/presentaciones/JDARE10-07.pdf>.

**TOVAR, C.** *Framework de programación de páginas web.* [en línea]. [Consulta: 8 julio 2019]. Disponible en: <https://es.slideshare.net/GeraldynDeSousa/framework-30197256>.

**VALBUENA, Á.** *Guía comparativa de frameworks para los lenguajes html 5, css y javascript para el desarrollo de aplicaciones web.* [en línea]. [Consulta: 8 julio 2019]. Disponible en: <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/4577/00676V865.pdf?sequence=1&isAllowed=y>.

**XALAMBRÍ, Sergio.** *Ventajas y desventajas de los pre-procesadores de CSS.* [Blog]. [Consulta: 9 julio 2019]. Disponible en: <https://medium.com/@sergiodxa/ventas-y-desventajas-de-los-pre-procesadores-de-css-6528fdac9926>.