



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

**“DESARROLLO DE UN SISTEMA WEB PARA EL CONTROL DE
VENTA Y CRIANZA DE CERDOS EN LA GRANJA PORCINA
“PLATANILLOS””**

TRABAJO DE TITULACIÓN

TIPO: PROYECTO TÉCNICO

Para optar al grado académico de:

INGENIERA EN SISTEMAS INFORMÁTICOS

AUTORA: KAREN ESTEFANÍA PARDO SARANGO

TUTOR: DR. JULIO ROBERTO SANTILLÁN CASTILLO

Riobamba-Ecuador

2019

©2019, Karen Estefanía Pardo Sarango

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

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

El Tribunal del Trabajo de Titulación certifica que: El Trabajo de investigación: “Desarrollo de un sistema web para el control de venta y crianza de cerdos en la granja porcina “PLATANILLOS””, de responsabilidad de la señorita: Karen Estefanía Pardo Sarango, ha sido minuciosamente revisado por los miembros del Tribunal, quedando autorizada su presentación.

NOMBRES	FIRMAS	FECHA
Dr. Julio Santillán Castillo	_____	_____
VICEDECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA		
Ing. Patricio Moreno Costales	_____	_____
DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS		
Dr. Julio Santillán Castillo	_____	_____
DIRECTOR DE TESIS		
Ing. Patricio Moreno Costales	_____	_____
MIEMBRO DEL TRIBUNAL		

Yo, Karen Estefanía Pardo Sarango soy responsable de las ideas, doctrinas y resultados expuestos en este trabajo de titulación y el patrimonio intelectual del trabajo de titulación pertenece a la Escuela Superior Politécnica de Chimborazo.

Karen Estefanía Pardo Sarango

DEDICATORIA

La presente tesis va dedicada a mis queridos padres Carmita Sarango y Vicente Pardo ya que ellos siempre me han dado su amor y apoyo incondicional para poder culminar esta hermosa etapa de mi vida, además me han enseñado a ser una persona de bien y a valorar cada cosa importante en mi vida.

A mi amado esposo José Luis Neira por siempre estar conmigo en las buenas y en las malas, brindándome su apoyo cuando más lo he necesitado, dándome su amor incondicional a pesar de todos los momentos difíciles que hemos vivido, por eso y mucho más sé que nos esperan mejores momentos juntos. Te amo.

A mi increíble y hermoso hijo Ezequiel Neira Pardo, que ha sido la fuente de inspiración para superarme día tras día y el que me dio la fuerza necesaria para no rendirme jamás y poder luchar por un futuro mejor.

A mis hermanos Cristian, Patricio y Hermes Pardo por siempre brindarme su apoyo incondicional a la distancia y darme su cariño incondicional.

Karen Estefanía

AGRADECIMIENTO

Agradezco principalmente a mi querido Dios, porque él siempre me dio la fortaleza y bendición para seguir adelante, a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO por permitirme cumplir una meta más en mi vida y a sus docentes por haber compartido sus conocimientos y así poder formarme como una buena profesional con grandes valores.

Agradezco a mis Padres por apoyarme siempre incondicionalmente y enseñarme a no rendirme pese a todos los obstáculos que se me han presentado, llenándome de grandes valores.

Agradezco a mi amado esposo y a mi hijo por hacer feliz cada día de mi vida y por estar en las buenas y en las malas junto a mí, gracias amor por no dejar que me rinda.

Al Doctor Julio Santillán agradezco el apoyo brindado con sus conocimientos guiándome para así poder culminar esta etapa de mi vida.

Karen Estefanía

TABLA DE CONTENIDO

RESUMEN	xiv
SUMARY	xv
INTRODUCCIÓN	1
CAPÍTULO I	5
1 MARCO TEÓRICO REFERENCIAL.....	5
1.1 Control venta y crianza de cerdos	5
1.2 Herramientas utilizadas en el desarrollo de la aplicación	6
1.2.1 Entorno de Desarrollo Integrado (IDE)	6
1.2.2 Lenguaje de Programación	9
1.2.3 Patrón de Diseño	13
1.2.4 Framework.....	16
1.2.5 Sistema Gestor de Base de Datos (SGBD).....	25
1.2.6 Metodología de desarrollo	30
1.2.7 Calidad del Software.....	34
CAPÍTULO II	38
2 MARCO METODOLÓGICO	38
2.1 Tipo de investigación	38
2.2 Población y muestra	38
2.3 Técnicas de investigación	38
2.4 Metodología Scrum para el desarrollo del sistema	39
2.4.1 Fase de planificación.....	39
2.4.2 Fase de desarrollo.....	46
2.4.3 Fase de cierre.....	68
CAPÍTULO III	70
3 RESULTADOS Y DISCUSIÓN	70
3.1 Resultado de medición de indicadores.....	70
3.1.1 Evaluación de la Usabilidad	70
3.1.2 Evaluación de la Eficiencia.....	81
CONCLUSIONES	86
RECOMENDACIONES	87
BIBLIOGRAFÍA	
ANEXOS	

ÍNDICE DE ABREVIATURAS

JAVA EE	Java Platform, Enterprise Edition
JSF	Java Server Faces
IDE	Entorno de Desarrollo Integrado
BD	Base de Datos
ISO	Organización Internacional de Normalización
MVC	Modelo Vista Controlador
HT	Historia técnica
HTTP	Protocolo de transferencia de hipertexto
HTML	Lenguaje de marcas de hipertexto
HU	Historia de Usuario
HW	Hardware
PDF	Formato de documento portátil
IEC	Comisión Electrotécnica Internacional
SW	Software
SQL	Lenguaje de consulta estructurada

ÍNDICE DE TABLAS

Tabla 1-2: Método de estimación T-Shirt	40
Tabla 2-2: Product Backlog - Historias de Usuario	40
Tabla 3-2: Product Backlog - Historias Técnicas	41
Tabla 4-2: Personas y roles	42
Tabla 5-2: Tipos y roles de usuario.....	42
Tabla 6-2: Sprint Backlog.....	43
Tabla 7-2: Historia Técnica 01.....	48
Tabla 8-2: Tarea de Ingeniería 01 de la HT_01	48
Tabla 9-2: Prueba de aceptación 01	48
Tabla 10-2: Identificación de riesgos.....	49
Tabla 11-2: Probabilidad del riesgo	49
Tabla 12-2: Impacto del riesgo	50
Tabla 13-2: Exposición del riesgo	50
Tabla 14-2: Impacto probabilidad.....	50
Tabla 15-2: Priorización de riesgos	50
Tabla 16-2: Priorización de riesgos 1	51
Tabla 17-2: Priorización de riesgos 2	51
Tabla 18-2: Priorización de riesgos 3	52
Tabla 19-2: Priorización de riesgos 4	53
Tabla 20-2: Priorización de riesgos 5	53
Tabla 21-2: Priorización de riesgos 6	54
Tabla 22-2: Priorización de riesgos 7	54
Tabla 23-2: Priorización de riesgos 8	55
Tabla 24-2: Historia Técnica 02.....	56
Tabla 25-2: Tarea de Ingeniería 01 de la HT_02	56
Tabla 26-2: Prueba de aceptación 01	56
Tabla 27-2: Historia Técnica 03.....	58
Tabla 28-2: Tarea de Ingeniería 01 de la HT_03	58
Tabla 29-2: Prueba de aceptación 01	58
Tabla 30-2: Historia Técnica 04.....	59
Tabla 31-2: Tarea de Ingeniería 01 de la HT_04	59
Tabla 32-2: Prueba de aceptación 01	60
Tabla 33-2: Tarea de Ingeniería 02 de la HT_04	60
Tabla 34-2: Prueba de aceptación 01	60
Tabla 35-2: Historia Técnica 05.....	62
Tabla 36-2: Tarea de Ingeniería 01 de la HT_05	62
Tabla 37-2: Prueba de aceptación 01	62
Tabla 38-2: Historia Técnica 06.....	66
Tabla 39-2: Tarea de Ingeniería 01 de la HT_06	66
Tabla 40-2: Prueba de aceptación 01	66
Tabla 41-2: Historia Técnica 07.....	67
Tabla 42-2: Tarea de Ingeniería 01 de la HT_07	67
Tabla 43-2: Prueba de aceptación 01	68
Tabla 1-3: Tipo de escala de medición de Likert.....	70
Tabla 2-3: Tipo de escala de medición de Likert a la inversa.....	71

Tabla 3-3: Cálculos para obtener la escala de Usabilidad.....	71
Tabla 4-3: Escala de Usabilidad.....	71
Tabla 5-3: Facilidad del manejo del sistema.....	72
Tabla 6-3: Comprensión de elementos en la interfaz	72
Tabla 7-3: Información de ubicación del usuario en el sistema	73
Tabla 8-3: Facilidad de uso del sistema	73
Tabla 9-3: Complejidad del sistema.....	74
Tabla 10-3: Mensajes de error mostrados por el sistema al usuario.	74
Tabla 11-3: Integración de las funciones del sistema	75
Tabla 12-3: Sencillez al buscar y filtrar información dentro del sistema.....	75
Tabla 13-3: Consistencia en el diseño del sitio web en todas las secciones	76
Tabla 14-3: Combinación de colores en el diseño del sistema	76
Tabla 15-3: Comprensibilidad en el sistema.....	77
Tabla 16-3: Facilidad de aprendizaje	78
Tabla 17-3: Operabilidad del sistema	78
Tabla 18-3: Atractivo del sistema	79
Tabla 19-3: Total de preguntas para medir la usabilidad del sistema.	80
Tabla 20-3: Escala de Eficiencia.....	81
Tabla 21-3: Procesos tradicionales en la granja porcina.....	81
Tabla 22-3: Tiempo en los procesos tradicionales de los usuarios que administran la granja... ..	82
Tabla 23-3: Tiempo en los procesos que realizan los usuarios utilizando el sistema.	83
Tabla 24-3: Comparación de los tiempos con y sin el uso del sistema.....	84

ÍNDICE DE FIGURAS

Figura 1-1: IDE Netbeans 8.2	7
Figura 2-1: Tipos de lenguajes de programación.....	9
Figura 3-1: Lenguaje de programación Java.....	11
Figura 4-1: Elementos de Java.....	12
Figura 5-1: Estructura del Modelo – Vista - Controlador.....	14
Figura 6-1: Tipos de Framework	16
Figura 7-1: Primefaces para JSF.....	20
Figura 8-1: Dispositivos compatibles con Bootstrap.....	22
Figura 9-1: Estructura de Bootstrap.....	24
Figura 10-1: Estructura de Bootstrap.....	24
Figura 11-1: Logo de PostgreSql.....	27
Figura 12-1: Funcionamiento de la metodología Scrum.....	30
Figura 13-1: Ciclo de desarrollo Scrum.....	33
Figura 1-2: Arquitectura del sistema	61
Figura 2-2: Cabecera de la página principal del sistema.	63
Figura 3-2: Cuerpo de la página principal del sistema.	64
Figura 4-2: Pie de la página principal del sistema.	64
Figura 5-2: Cabecera de la página principal de ingreso al sistema.....	65
Figura 6-2: Cuerpo de la página principal de ingreso al sistema.	65
Figura 7-2: Pie de la página principal de ingreso al sistema.....	65

ÍNDICE DE GRÁFICOS

Gráfico 1-2: Burn Down Chart	69
Gráfico 1-3: Usabilidad del sistema.....	80
Gráfico 2-3: Comparación de los tiempos estimados	84
Gráfico 3-3: Comparación total de los tiempos promedio estimados	85

ÍNDICE DE ANEXOS

ANEXO A: Planificación del sistema – Diagrama de Gantt

ANEXO B: Diseño de la Base de Datos

ANEXO C: Diccionario de Datos

ANEXO D: Historias de Usuarios

ANEXO E: Modelo de Entrevista

ANEXO F: Modelo de Medición del Tiempo

ANEXO G: Resultados de la Eficiencia con el uso del sistema

RESUMEN

En el presente trabajo de titulación se desarrolló un sistema de venta y crianza de cerdos para la granja “Platanillos” aplicando la norma ISO/IEC 9126 para determinar el nivel de usabilidad del mismo, se realizó un análisis de la situación actual que lleva la granja en los procesos de venta y crianza de cerdos. Para el desarrollo del mismo se utilizó el lenguaje de programación Java, la herramienta de desarrollo Netbeans utilizando el framework Java Server Faces, Primefaces y Bootstrap aplicando la arquitectura Modelo, Vista, Controlador y utilizando PostgreSql como gestor de base de datos, se utilizó las técnicas de recopilación de datos como la entrevista y las fichas de observación, además, se aplicó la metodología de desarrollo SCRUM para dar seguimiento al desarrollo del sistema e ir dando entrega de los requerimientos dados por el cliente. Para evaluar la usabilidad del sistema se realizaron pruebas de funcionamiento con los usuarios y se les aplicó una entrevista, en las cuales un 93.75% manifestaron estar satisfechos con la usabilidad del sistema; asimismo, para medir la eficiencia del software se utilizaron entrevistas y se obtuvo información de los tiempos de los procesos de forma tradicional, y se les comparó con los tiempos de los procesos utilizando el sistema; el resultado obtenido es que el sistema reduce el tiempo en un 91.02%, La aplicación fue aceptada en un 100% y es el resultado de haber realizado un desarrollo de software aplicando las nuevas metodologías de programación. En conclusión, la aplicación permite generar de forma fácil y rápida los reportes de ventas realizadas y la verificación de creación de salas entre fechas con óptimos tiempos de proceso. Se recomienda realizar el estudio minucioso y una comprensión técnica avanzada del funcionamiento de la aplicación en cuanto a su integración con otras herramientas de trabajo.

PALABRAS CLAVE: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <INGENIERÍA DE SOFTWARE>, < APLICACIÓN WEB>, < SISTEMA GESTOR DE BASE DE DATOS (SGBD)>, <PRIMEFACES (FRAMEWORK)>, < BOOTSTRAP (FRAMEWORK)>, < JSF (JAVA SERVER FACES)>, <METODOLOGÍA ÁGIL SCRUM>, <NORMA ISO/IEC 9126-2>, <EVALUACIÓN DE LA EFICIENCIA>, <EVALUACIÓN DE LA USABILIDAD>.

SUMMARY

In this research work was developed a system of sale and pig breeding for "Platanillos" farm by applying the norm ISO/IEC 9126 to determine the level of usability of it. An analysis of the current situation was carried out that takes the farm in the processes of selling and pig breeding. The Java programming language was used to develop it, the Netbeans development tool using the Java Server Faces framework, Primefaces and Bootstrap applying Vista architecture Model, Controller and using PostgreSQL as the database manager, using the techniques data collection such as the interview and the observation sheets. In addition, the SCRUM development methodology was applied to monitor the development of the system and to give delivery of the requirements given by the client. To evaluate the usability of the system, functional tests were performed with the users and an interview was applied, in which 93.75% said they were satisfied with the usability of the system; Likewise, to measure the efficiency of the software, interviews were used and information was obtained of the times of the processes in a traditional way. They were compared with the times of the processes using the system; the result obtained is that the system reduces the time by 91.02%. The application was accepted 100% and as a result of conducting a software development applying the new programming methodologies. In conclusion, the application allows easy and quick generation of sales reports and verification of creation of rooms between dates with optimal process times. It is recommended to carry out the meticulous study and an advanced technical understanding of the operation of the application in terms of its integration with other work tools.

KEYWORDS: <TECHNOLOGY AND SCIENCE OF ENGINEERING>, <SOFTWARE ENGINEERING>, <WEB APPLICATION>, <SYSTEM DATABASE MANAGER (SGBD)>, <PRIMEFACES (FRAMEWORK)>, <BOOTSTRAP (FRAMEWORK)>, < JSF (JAVA SERVER FACES)>, <ÁGIL SCRUM METHODOLOGY>, <ISO NORM / IEC 9126-2>, <EVALUATION OF EFFICIENCY>, <USABILITY EVALUATION >.

INTRODUCCIÓN

La tecnología con el pasar del tiempo muestra un gran avance; por medio de ésta han progresado los diferentes campos, tales como: la ciencia, la medicina, el deporte entre otros. Asimismo, el asombroso avance tecnológico ha llevado a que todos los procesos de cualquier ámbito sean automatizados; con el fin de que el trabajo sea más rápido y productivo.

La expansión del internet donde la información es accesible desde cualquier lugar, sea dentro o fuera del país, hace que la información sea compartida entre las partes interesadas, de tal manera que éstas tengan total acceso en cualquier momento.

La necesidad de las personas provoca que exista un cambio de las aplicaciones tradicionales de escritorio, hacia las aplicaciones web; por lo tanto, las aplicaciones web se han convertido en un medio importante de interacción entre las empresas y las personas. Su fácil implementación, han hecho que sean prácticamente esenciales en el comercio electrónico.

Las granjas porcinas instaladas en los últimos años en el país, se han convertido en un gran negocio, puesto que, la inversión es mínima y muestra una gran rentabilidad. Por consiguiente, la economía de las personas mejora considerablemente, y atrae a más inversionistas a este negocio. Por lo anterior, se han realizado trabajos de investigación referentes a la automatización de los procesos productivos de crianza y venta de cerdos de las granjas.

La granja porcina “Platanillos” se inició en la ciudad de Piñas, como una granja de explotación reproductiva para la comercialización de carne. El proceso productivo se realiza de manera tradicional, apoyándose en la planificación y control por los responsables de áreas, con la ayuda de formatos impresos predefinidos; de tal manera que exista un registro de las actividades diarias. Esto provoca gastos en la granja, haciendo uso de recursos excesivos por falta de planificación y por no detectar fallas en la producción a tiempo.

En vista a la problemática que existe en la granja porcina “PLATANILLOS”, en cuanto a los gastos y tiempos excesivos para el control de venta y crianza de cerdos, se cree conveniente desarrollar un sistema web; el mismo que ayudará a organizar la información dispersa, reducir el tiempo y los costos de la empresa.

Para ello se plantea un proyecto que permita realizar un seguimiento de los eventos durante la producción, facilitando la ejecución de medidas correctivas en el momento oportuno y que contribuya a la obtención de altos rendimientos de producción y a la disminución de los gastos.

Las herramientas que se emplean para la realización del proyecto son: lenguaje de programación java, Netbeans como IDE de desarrollo, PostgreSQL como gestor de base de datos y los Framework Primefaces y Bootstrap, los cuales ayudan a disminuir el tiempo y el esfuerzo en el desarrollo del sistema web.

Para dar seguimiento al desarrollo del sistema web “Platanillos”, se utiliza la metodología ágil SCRUM, mediante la cual se definen los requerimientos tanto funcionales como no funcionales. Esta metodología prioriza los requerimientos en el Product Backlog y posteriormente en el Sprint Backlog, con el fin de asignar los recursos necesarios para el desarrollo de cada uno de los requerimientos.

Las herramientas tienen como finalidad disminuir los tiempos de cada fase, y mejorar los resultados obtenidos para dar propuestas ideales al cliente.

A continuación, se detallan algunos de los problemas encontrados en la granja porcina.

El seguimiento de la crianza de los animales no se está realizando adecuadamente.

No se realiza un registro oportuno de la fecha de nacimiento del animal, hasta el tiempo que debe permanecer con su madre, para luego pasar al proceso de venta; además, no se registra un seguimiento adecuado de las fechas y de los cambios de la sala oportunamente.

La evolución de los cerdos no es controlada adecuadamente.

No se conoce con exactitud el número de animales que se maneja en cada etapa de crianza, tampoco se maneja información adecuada sobre la evolución del animal en lo referente a: edades, medicación y alimentación. Como consecuencia las raciones de alimentos a dosificar son insuficientes o excesivas, perjudicando el desarrollo corporal de los animales y el presupuesto para alimentos y medicinas.

El control de nacimiento de los cerdos es insuficiente.

Cuando se da un parto se registra en fichas que guardan datos de la marrana y de los lechones recién nacidos, sin llevar un control de la evolución y desarrollo de la etapa de maternidad. Por consiguiente, no es posible una asignación de sala adecuada.

La disponibilidad de los alimentos para los cerdos no está debidamente controlada.

No existe un control, registro o inventario actualizado de los alimentos utilizados para la distribución en áreas de producción, por lo que se corre el riesgo que existan demandas no atendidas y los animales no cuenten con un desarrollo normal.

El control de las ventas de los cerdos se realiza de forma tradicional.

Cuando los cerdos tienen la edad y peso ideal para venta, son entregados para la comercialización. La salida de cada embarque es registrada en fichas o guías de salida y no se lleva un control detallado de: los precios de venta, la fecha de salida y quien realiza la compra.

La adquisición de los productos medicinales y alimenticios no son controlados.

Cuando la granja porcina adquiere productos, el ingreso al almacén se hace de forma manual, sin la actualización exacta de la cantidad de cada producto existente.

No se lleva un control de las salas que existen en la granja.

La granja cuenta con diferentes salas las cuales no están administradas adecuadamente, no se conoce si están con o sin cerdos, además, se desconoce el tipo de sala, la fecha de creación o cierre de las mismas.

No se conoce de forma exacta las ventas que se realizaron entre determinadas fechas.

No se conoce con exactitud qué ventas se realizaron entre determinadas fechas, dado que, toda venta se realiza de manera tradicional y escrita en papel; a veces las facturas o notas de venta se pierden o se dañan, por lo anterior, no se puede conocer de forma exacta las ventas que se han realizado entre fechas.

El control de hembras o machos en cada sala es desconocido.

Al momento de ingresar lechones a una nueva sala no se controla la cantidad de hembras o machos, solo pasan lechones a la nueva sala y se termina el proceso de ingreso de los animales.

El proyecto planteado permite realizar un seguimiento de los eventos que se producen durante la crianza y venta de cerdos, facilitando la ejecución de medidas correctivas en el momento oportuno. Esto, contribuye la obtención de altos rendimientos de producción y ventas. Además, permite planificar las actividades y las necesidades de la granja, así como también, mejorar el control y la eficiencia de ésta, mediante el uso de indicadores de evaluación.

En la actualidad existen softwares en Estados Unidos de gestión y administración porcina, uno de ellos es “Porcitech” (AGRITEC), que optimiza el rendimiento de una granja y aumenta la productividad. Además, proporciona informes para la toma de decisiones mediante una entrada de datos rápida y la muestra de un sencillo reporte.

En la Escuela Superior Politécnica de Chimborazo (ESPOCH), un trabajo de investigación desde el 2018, muestra que en el Anexo 3 se enmarca en la Tabla 2. Líneas y programas institucionales

de investigación en las áreas de UNESCO de ciencias tecnológicas en las LÍNEAS de las Tecnologías de la información, comunicación y procesos industriales; con el PROGRAMA para el desarrollo de aplicaciones de software, hardware y telecomunicaciones; En la INSERCIÓN DENTRO DE LOS OBJETIVOS DEL MILENIO con el objetivo 8: Fomentar una asociación mundial para el desarrollo junto con el PLAN DEL DESARROLLO DEL BUEN VIVIR en los objetivos 3 y 7. En el cual el Objetivo.3: Mejorar la calidad de vida de la población y el Objetivo. 7: Garantizar los derechos de la naturaleza y promover la sostenibilidad ambiental, territorial y global. (Fiallos y ESPOCH, 2018)

Objetivos

Objetivo General

Desarrollar un sistema web para el control de venta y crianza de cerdos en la granja porcina “PLATANILLOS”.

Objetivos Específicos

- Analizar y determinar los requerimientos técnicos y tecnológicos para el control de venta y crianza de cerdos de la granja porcina “Platanillos”
- Diseñar, desarrollar e implementar un sistema web para el control de venta y crianza de cerdos de la granja aplicando la metodología Scrum.
- Determinar el grado de usabilidad y rendimiento del sistema para verificar la funcionalidad del sistema.

En el presente trabajo de titulación se definen tres capítulos distribuidos de la siguiente manera: el Capítulo uno presenta el marco teórico, el mismo que contiene las definiciones de las herramientas para el desarrollo del sistema así como también: conceptos básicos y específicos utilizados para el desarrollo del mismo, en el Capítulo dos tenemos el marco metodológico el mismo que contiene los métodos y técnicas de investigación, y en el Capítulo tres tenemos el análisis de la recolección de datos para los resultados y conclusiones.

CAPÍTULO I

1 MARCO TEÓRICO REFERENCIAL

En el presente capítulo se muestra el marco teórico, en el cual se describen las herramientas utilizadas para el desarrollo del sistema, así como también, se definen conceptos básicos y específicos complementarios al desarrollo del sistema para control de venta y crianza de cerdos en la granja porcina “Platanillos”.

1.1 Control venta y crianza de cerdos

El control de venta y crianza de cerdos de la granja porcina tiene como objetivo principal administrar las actividades que están enfocadas al bienestar de los cerdos en cuanto a su crianza, permitiendo contar con información segura, confiable y disponible, para mejorar la gestión de la granja. (Egea, 2017)

Características de un control de venta y crianza

- Realizar Ingreso de cerdos.
- Realizar ingreso de alimentos para los cerdos.
- Realizar ingreso de medicinas para los cerdos.
- Administrar la alimentación.
- Administrar la medicación.
- Administrar los cerdos por etapas.
- Generar informes sobre los cerdos.
- Generar informes sobre los alimentos y medicamentos.
- Generar informes sobre venta de cerdos. (Muñoz Armas, 2011)

Beneficios del control de venta y crianza

- Almacenamiento de la información ordenada.
- Acceso rápido y seguro a la información de la granja, cerdos y ventas.
- Reducción de tiempo en cuanto a procesos de crianza.
- Generación de informes de manera más fácil y rápida. (Salesforce, 2000)

1.2 Herramientas utilizadas en el desarrollo de la aplicación

1.2.1 Entorno de Desarrollo Integrado (IDE)

Un entorno de desarrollo integrado, es un programa compuesto por un conjunto de herramientas para un programador, donde se dedica exclusivamente a uno o varios lenguajes de programación. Además, es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. (Valdiviezo Serrano y Guacho Minta, 2012, p. 61)

Componentes de un Entorno de Desarrollo Integrado (IDE)

- Editor de texto.
- Compilador.
- Intérprete.
- Herramientas de automatización.
- Depurador.
- Posibilidad de ofrecer un sistema de control de versiones.
- Factibilidad para ayudar en la construcción de interfaces gráficas de usuarios. (Valdiviezo Serrano y Guacho Minta, 2012, p. 62)

Ejemplos de entornos de desarrollo integrados (IDE)

- C++ Builder y Turbo C++ Explorer de Borland (ahora, de CodeGear).
- Clarion.
- Code: : Blocks.
- CodeWarrior.
- Delphi y Turbo Delphi de Borland (ahora de CodeGear).
- Eclipse.
- JBuilder de Borland (ahora de CodeGear).
- JDeveloper de Oracle_Corporation.
- JCreator.
- KDevelop (QT, pensado para escritorio KDE).
- Lazarus.
- MS Visual Studio .NET y Visual Studio Express de Microsoft.
- Netbeans.

- Turbo Pascal de Borland.
- Watcom C/C++ Compiler y OpenWatcom.
- Xcode de Apple. (Valdiviezo Serrano y Guacho Minta, 2012, p. 63)

Se elige un IDE para trabajar con Java, dado que es muy importante y permite crear un conjunto de aplicaciones de apoyo a los desarrolladores, además, provee un entorno de trabajo muy amigable para la mayoría de lenguajes de programación y puede funcionar como un sistema en tiempo de ejecución, el cual permite utilizar el lenguaje de programación en forma interactiva, sin la necesidad de trabajo orientado a los archivos de texto.

1.2.1.1 Netbeans 8.2

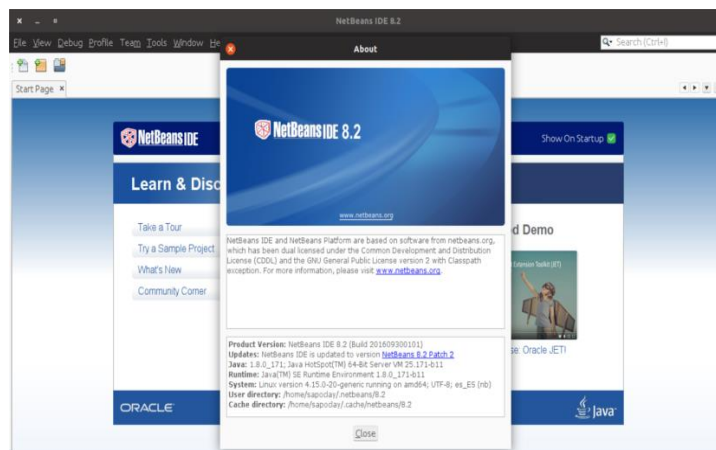


Figura 1-1: IDE Netbeans 8.2

Fuente: (Amoedo, 2018)

Netbeans IDE es un entorno de desarrollo creado por un Microsystems, herramienta en la que se puede escribir, compilar, depurar y ejecutar programas escritos en Java, aunque también puede servir para cualquier otro lenguaje de programación. En esta herramienta existe un gran número de módulos para ampliarla. Además, es un producto de uso libre, gratuito y no tiene restricciones de uso. (Arbeláez Salazar et al., 2011)

Netbeans 8.2 es un entorno de desarrollo integrado modular basado en estándares, escrito en el lenguaje de programación Java de código abierto con todas las funciones y plataforma de aplicaciones cliente enriquecida, se puede usar como un marco genérico para construir cualquier tipo de aplicación.

Los beneficios que ofrece la utilización de Netbeans a un programador son:

- Se pueden realizar todas las tareas asociadas a la programación como: editar el código, compilarlo, ejecutarlo y depurarlo.
- Simplifica tareas tediosas en proyectos.
- Asiste parcialmente en la escritura de código, aunque no libera del aprendizaje del lenguaje de programación.
- Ayuda en la navegación de las clases predefinidas en la plataforma.
- Aunque puede ser costoso su aprendizaje, los beneficios superan las dificultades. (Herrera Saltos y Vargas Guambo, 2016)

El IDE Netbeans es un reconocido entorno de desarrollo que está disponible para Windows, Mac, Linux y Solaris. Además, está formado por una plataforma de aplicación que permite a los desarrolladores crear con rapidez aplicaciones web, empresariales, de escritorio y para móviles utilizando la plataforma Java, así como: Java FX, PHP, Java Script y Ajax, C/C++, etc. El proyecto de Netbeans está apoyado por una comunidad dinámica de desarrolladores que ofrece documentación y recursos de forma exhaustiva, así como una amplia selección de complementos. (Ramos, 2017, p. 33)

Netbeans en su plataforma permite realizar varias actividades dentro del ambiente de desarrollo ya sea editar, compilar, ejecutar, depurar entre otras actividades, que ayudan a tener un mejor desarrollo del código. Además, ayuda a que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados *módulos*. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs y un archivo especial que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándoles nuevos módulos, debido a que éstos pueden ser desarrollados independientemente. (Valdiviezo Serrano y Guacho Minta, 2012, pp. 66–67)

La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiendo al desarrollador enfocarse en la lógica específica de la aplicación.

Las características de la plataforma Netbeans son:

- Administración de las interfaces de usuario (ej. menús, barras de herramientas, etc.).
- Administración de las configuraciones del usuario.
- Administración de ventanas.
- Administración del almacenamiento.

- Framework basado en asistentes (diálogos paso a paso). (Valdiviezo Serrano y Guacho Minta, 2012, p. 69)

Se decide utilizar Netbeans porque soporta el lenguaje de programación Java con el cual se va a programar, tiene asistentes para la creación y configuración de algunos proyectos, así mismo incluye algunos framework con los que se va a desarrollar el sistema. También se puede conectar a distintos sistemas de base de datos para poder acceder a la información necesaria.

1.2.2 Lenguaje de Programación



Figura 2-1: Tipos de lenguajes de programación

Fuente: (Pérez, 2016)

Un lenguaje de programación tiene su estructura con cierta base sintáctica y semántica, imparte distintas instrucciones a un programa de computadora. A la hora de establecer el origen del lenguaje de programación se tiene que hacer referencia, sin lugar a dudas, a Ada Lovelace que está considerada como la primera programadora de computadoras conocida en todo el mundo. De ahí se empezó hablar en su honor sobre el lenguaje de programación Ada. Y es que dicha figura llevó a cabo no sólo la manipulación de una serie de símbolos para una máquina del científico británico Charles Babbage sino también la consecución del establecimiento de las instrucciones necesarias para que un computador pudiera realizar una serie de cálculos iniciales. (Pérez Porto y Merino, 2009)

Inicialmente, los lenguajes de programación web eran lenguajes estáticos que hacían que las páginas web diseñadas fueran poco flexibles. Actualmente, gracias a la evolución de la tecnología y sobre todo de las exigencias de los usuarios han aparecido nuevos lenguajes dinámicos que manejan potentes bases de datos y que han permitido dar solución a estas exigencias, dando a los usuarios la manera de interactuar con las webs y entre ellos mismos. Ahora existe un buen número de lenguajes de programación que han ido apareciendo a medida que el internet ha ido evolucionando, y los requerimientos de los servidores y la velocidad de las conexiones han ido cambiando. (Pepe, 2017)

Existen lenguajes de programación en el mercado y algunos de ellos utilizan la denominación “Visual” en su nombre (Visual C, Visual Basic, etc.). La mayoría de estos lenguajes no son más que el 50% visual, ya que están basados en una combinación de objetos gráficos y de texto. Sin embargo, existen lenguajes de programación 100% gráficos, es decir, lenguajes que permiten programar y desarrollar una aplicación sin utilizar texto, y este tipo de lenguaje de programación es el que permite el desarrollo de los instrumentos virtuales. (Tisalema Paninboza y Muñoz Arias, 2012, p. 24)

Los objetivos específicos de los diferentes lenguajes de programación son:

- Facilidad de uso.
- Soluciones específicas como sistemas operativos, bases de datos, etc.
- Emular la construcción de hardware en los dispositivos que están formados por objetos.
- Reducción de los errores de mecanografía estricta. (Tisalema Paninboza y Muñoz Arias, 2012)

A continuación, un listado con los principales lenguajes de programación existentes:

- ADA.
- Basic.
- HTML.
- Java.
- Java script.
- Fortran.
- PHP.
- C y C++.
- ASP y ASP.NET.
- AJAX.
- Cobol.
- Java Server Pages.
- Pascal.
- Matlab.
- Perl.
- Ruby. (Pepe, 2017)

- Proporciona herramientas y librerías para que los programas se ejecuten en varias máquinas. (Herrera Saltos y Vargas Guambo, 2016, p. 23)

La Plataforma Java se compone de un amplio abanico de tecnologías, cada una de las cuales ofrece una parte del entorno de ejecución en tiempo real. Por ejemplo, los usuarios finales interactúan con la máquina virtual de Java y el conjunto estándar de bibliotecas. Asimismo, las aplicaciones Java pueden usarse de forma variada, como por ejemplo ser incrustadas en una página Web. Para el desarrollo de las aplicaciones, se utiliza un conjunto de herramientas conocidas como JDK (Java Development Kit). (Valdiviezo Serrano y Guacho Minta, 2012, p. 32)

La plataforma de java incluye:

- Edición Estándar (Standard Edition), o Java SE (antes J2SE)
- Edición Empresa (Enterprise Edition), o Java EE (antes J2EE)
- Edición Micro (Micro Edition), o Java ME. (Valdiviezo Serrano y Guacho Minta, 2012, p. 32)

Librerías de Java

Java ofrece un conjunto de librerías estándar, que contiene muchas de las funciones reutilizables disponibles en los sistemas operativos actuales. Las librerías tienen propósitos dentro de la plataforma, al igual que otras librerías estándar ofrecen al programador un conjunto bien definido de funciones para realizar tareas comunes, como: manejar listas de elementos y operar de forma sofisticada sobre cadenas de caracteres. Además, proporcionan una interfaz abstracta para tareas que son altamente dependientes del hardware de la plataforma destino y de su sistema operativo. Tareas como: manejo de las funciones de red o acceso a ficheros, suelen depender fuertemente de la funcionalidad nativa de la plataforma destino. Finalmente, no todas las plataformas soportan todas las funciones de una aplicación Java, ya que las librerías pueden emular funciones usando lo que esté disponible, o bien ofrecer un mecanismo para comprobar si una funcionalidad concreta está presente. (Valdiviezo Serrano y Guacho Minta, 2012, p. 33)

Java incluye un compilador y un intérprete. El compilador produce un código de bytes que se almacena en un fichero para ser ejecutado por un intérprete denominado máquina virtual de java. (Acosta Fiallos y Monar Borja, 2016, p. 7)



Figura 4-1: Elementos de Java

Realizado por: Karen Pardo. 2019

Reglas de codificación de Java

Existen ciertas reglas de codificación que son comúnmente utilizadas por los programadores como:

- Los nombres de las clases deben empezar con mayúscula.
 - Los atributos y métodos de las clases deben empezar con minúsculas y si están formadas por varias palabras, se escriben sin espacios y la primera letra de cada palabra en mayúscula.
 - Las constantes se escriben en mayúsculas.
 - Las instancias de las clases siguen la misma recomendación que los métodos y atributos.
- (Belmonte Fernández, 2004, p. 4)

Tipos de datos en Java

- Datos primitivos.
- Datos referencia.
- Recolector de basura.
- Declaración de variables.
- Ámbito de las variables.
- Operadores.
- Arrays.
- Cadenas de caracteres. (Belmonte Fernández, 2004)

Después de investigar acerca de Java se decide utilizarlo, dado que es un lenguaje de programación muy práctico que ayuda al acceso y gestión de los sistemas de archivos, gráficos y sonidos. Además, es multiplataforma y es utilizado para crear aplicaciones web.

1.2.3 Patrón de Diseño

El patrón de diseño es el esqueleto o la base para la búsqueda de soluciones a los problemas comunes que se dan en el desarrollo de software. Otra de sus características es la reutilización de código. (Rios Luna, 2014, p. 11)

Los patrones de diseño representan una evolución muy importante en la abstracción y reutilización del software, la abstracción representa la forma que tienen los desarrolladores para resolver problemas complejos dividiéndolos en otros más simples. (Rios Luna, 2014, p. 12)

Clasificación de los patrones por tipos y niveles:

- Tipos
 - De creación de objetos.
 - De comportamiento, que coordinan la interacción funcional entre los objetos.
 - Estructurales, que gestionan relaciones estáticas y entre los objetos.
 - De sistema, que gestionan la interacción a nivel de todo el sistema. (Rios Luna, 2014, p. 12)
- Niveles
 - Clase única: Donde el patrón se aplica a una única clase.
 - Arquitectónico: Donde el patrón se utiliza para coordinar todas las acciones.
 - Componente: Donde el patrón implica un grupo de clases. (Rios Luna, 2014, p. 12)

Un patrón de diseño es una herramienta muy útil que ayuda a desarrollar la aplicación web de manera fácil y apropiada, reutilizando código y acoplándolo. Por tal motivo se selecciona un patrón de diseño.

1.2.3.1 Modelo Vista Controlador (MVC)

El modelo - vista – controlador es un patrón de arquitectura software, que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario, y el módulo encargado de gestionar eventos y comunicaciones. Además, define componentes para representar la información y la interacción del usuario. Este patrón se basa en ideas de reutilización de código y separación de conceptos, características para facilitar la tarea de desarrollo de aplicaciones y de mantenimiento. (EGAS CLAVIJO, 2015, p. 11)

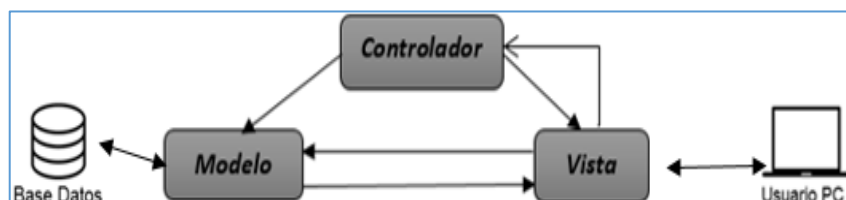


Figura 5-1: Estructura del Modelo – Vista - Controlador

Realizado por: Karen Pardo

Componentes del MVC

- *Modelo:* Representa la información con la que el sistema trabaja, por lo tanto, gestiona cada uno de los accesos a dicha información como: consultas, actualizaciones,

implementaciones de acceso descritas en las especificaciones de la aplicación y la lógica de negocio. Además, envía a la “vista” información que se solicita para que ésta sea mostrada al usuario; así mismo, a través del controlador se envían peticiones de acceso o manipulación de información. (EGAS CLAVIJO, 2015, p. 12)

- *Vista:* Muestra el Modelo (la información y la lógica de negocio) en un formato apropiado para poder interactuar con el usuario, por lo tanto, se requiere de dicho Modelo para mostrar la información. (EGAS CLAVIJO, 2015, p. 12)
- *Controlador:* Responde a eventos o acciones de usuario e invoca lo que son las peticiones al modelo cuando se le hace alguna solicitud sobre la información. También, puede enviar comandos a la Vista cuando se solicita algún cambio en la forma en que se presenta el Modelo, por lo tanto, se podría decir que el Controlador hace de intermediario entre la Vista y el Modelo. (EGAS CLAVIJO, 2015, p. 12)

Características del MVC

- El modelo es la capa donde se trabaja con todos los datos, además contiene todas las funciones con las que se puede acceder a las tablas y hacer las correspondientes inserciones, actualizaciones, eliminaciones, etc.
- La vista, es la capa que contiene todos los formularios que son visualizados dentro de las interfaces de usuario, permite con ello la interacción entre el usuario y la aplicación.
- El controlador es la capa que sirve como enlace entre la vista y el modelo, respondiendo a todos los mecanismos que puedan requerirse para implementar las necesidades de la aplicación. Además, contiene el código que es necesario para responder a todas las acciones que se solicitan en la aplicación como: visualizar un elemento, realizar una venta o búsqueda de información, etc. (Quingatuña Moreano y Nogales Guerrero, 2016, p. 29)

Funcionamiento del Modelo – Vista - Controlador (MVC)

- El usuario final realiza una petición
- El controlador captura dicho evento
- Se hace la llamada al modelo
- El modelo se encarga de interactuar con la base de datos (BD), sea de forma directa o con una capa de abstracción y el servicio web (Web Service) retornará la información al controlador.
- El controlador recibe la información y la envía a la vista.

- La vista procesa esta información desde el enfoque, creando una capa de abstracción para la lógica de negocio y el diseño de interfaz gráfica. (Yuca García y Tijero Fuentes, 2017, p. 30)

El Modelo – Vista – Controlador (MVC), es el patrón de diseño seleccionado que ayuda a separar los datos, la lógica de negocio y la interfaz de usuario dentro de la aplicación.

1.2.4 Framework



Figura 6-1: Tipos de Framework

Fuente: (Cameron, 2016)

El propósito de un framework es normalizar y estructurar el código del sistema, facilitando un esquema, un patrón o un esqueleto para el desarrollo y/o la implementación de aplicaciones. El uso de un framework para cualquier tipo de desarrollo, reduce el tiempo de elaboración e implementación y ayuda hacer un trabajo escalable según las características del mismo. (Villalobos et al., 2010)

Actualmente, con la aparición de los framework el tiempo de desarrollo de una aplicación se reduce considerablemente, debido a la ayuda de estas nuevas herramientas. Así mismo, teniendo como aliadas a las metodologías ágiles de desarrollo, se puede prototipar aplicaciones en menos tiempo y terminar su desarrollo en un par de semanas. También, se pueden encontrar varios framework para el desarrollo de aplicaciones médicas, de juegos, de visión por computador, y para cualquier ámbito que se desee. (Valdiviezo Serrano y Guacho Minta, 2012, p. 36)

Definición

Es un conjunto de componentes para el desarrollo de software, tales como: las clases en java, los descriptores y los archivos de configuración en XML; todos estos componentes forman un diseño reutilizable que facilita y agiliza el desarrollo de sistemas web. Además, tiene como objetivo

acelerar el proceso de desarrollo del sistema, reutilizar el código ya existente y promover el buen uso de los patrones de diseño. (Valdiviezo Serrano y Guacho Minta, 2012)

Tipos de Framework Web

Existen varios tipos de framework para el desarrollo de un sistema web, tales como: Java Server Faces que es orientado a la interfaz de usuario, Cocoon orientado a las aplicaciones para la publicación de documentos, Strut orientado a la parte de control de eventos, Bootstrap orientado a la creación de interfaces que se adaptan al tamaño del dispositivo que se utiliza; así entre otros framework para diferentes áreas. (Valdiviezo Serrano y Guacho Minta, 2012, p. 38)

Características de los framework

- Tiene acceso a datos.
- Existe abstracción de URLs y sesiones.
- Cuenta con controladores para gestionar eventos como introducción de datos, mediante un formulario o acceso a una página.
- Internacionalización.
- Existe separación entre el diseño y el contenido.
- Autenticación y control de acceso. (Valdiviezo Serrano y Guacho Minta, 2012, p. 39)

Un framework permite separar en tres capas a una aplicación, estas capas son:

- *La lógica de presentación.* - Es la que administra las interacciones entre el usuario y el software.
- *La Lógica de datos.* - Es la que permite el acceso a un agente de almacenamiento persistente.
- *La lógica de dominio o de negocio.* - Es la que manipula los modelos de datos de acuerdo a los comandos recibidos desde la presentación. (Villalobos et al., 2010)

Un framework tiene una estructura conceptual y tecnológica que evita escribir código repetitivo, para desarrollar un sistema en menos tiempo; contiene patrones de desarrollo que separan los datos y la lógica de negocio de la interfaz de usuario. Por tal motivo se decide utilizar un framework.

1.2.4.1 Java server faces (JSF)

Java Server Faces (JSF), es un framework basado en el patrón MVC (Modelo Vista Controlador) para aplicaciones Java basadas en la web, el cual simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. Un framework es una estructura de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente suelen incluir soporte de programas, bibliotecas y un lenguaje interpretado para ayudar a desarrollar y unir los diferentes componentes de un proyecto. (Herrera Saltos y Vargas Guambo, 2016, p. 25)

Características

- Se ejecuta del lado del servidor
- La especificación JSF fue diseñada según los siguientes principios:
 - ✓ Amigable.
 - ✓ Independiente del protocolo y del dispositivo del cliente.
 - ✓ Posibilidad de utilizarse con tecnologías Java Server Pages (JSP).
 - ✓ Independencia de tecnologías JSP.
 - ✓ Escalable.
 - ✓ Posibilidad de utilizarse con navegadores Web actuales. (Herrera Saltos y Vargas Guambo, 2016, p. 25)
- Tiene componentes principales.
 - ✓ API de referencia para representar componentes de interfaz de usuario y manejar su estado, manejo de eventos, validación del lado del servidor y conversión de datos, además, permite definir las reglas de navegación entre páginas.
 - ✓ Librería de etiquetas JSP, personalizadas para dibujar componentes de interfaz dentro de una página. (Herrera Saltos y Vargas Guambo, 2016, p. 25)
- Los elementos pueden:
 - ✓ Conectar eventos generados en el cliente con código de la aplicación en el servidor web.
 - ✓ Mapear componentes UI a la página de datos del servidor web.
 - ✓ Construir un interfaz de usuario con componentes reutilizables y extensibles.
 - ✓ Grabar y restaurar el estado del interfaz de usuario más allá de la vida de las peticiones del servidor. (Herrera Saltos y Vargas Guambo, 2016, pp. 25–26)

Ventajas

- Ofrece una visible separación entre el comportamiento y la presentación.
- Se hace fácil de usar al aislar al desarrollador del API (Interfaz de Programación de Aplicaciones) de Servlet.
- Proporciona un sencillo modelo de programación para enlazar todas las piezas.
- Las APIs de la tecnología Java Server Faces se han creado directamente sobre el API Java Servlet. Esto permite hacer algunas cosas como: usar otra tecnología de presentación junto a JSP, crear componentes propios personalizados directamente desde las clases de componentes y generar salida para diferentes dispositivos.
- Java Server Faces ofrece una gran cantidad de componentes de código abierto para las funcionalidades que se necesiten. Además, también existe una gran cantidad de herramientas para el IDE en JSF al ser el estándar de JAVA.
- La tecnología JSF proporciona una rica arquitectura para manejar el estado de los componentes, procesar los datos, validar la entrada del usuario y manejar eventos.
- Ofrece una rápida adaptación para nuevos desarrolladores. (Valdiviezo Serrano y Guacho Minta, 2012, p. 57)

Estructura de las aplicaciones JSF

- **/Java Source:** Fuentes de clases Java.
- **/ant:** Scripts de creación.
- **/WebContent:** Archivos de la aplicación web usados por el servidor de aplicaciones.
- **/Web-INF:** Archivos utilizados como parte de la rutina de la aplicación web.
- **/pages:** Archivos jsp para la presentación de la interfaz. (Ferro y Sanchez, 2006)

Java Server Faces es un framework modelo - vista - controlador que proporciona un conjunto de componentes en forma de etiquetas definidas en xhtml; éstas ayudan a las interfaces de usuario mediante vistas, además agrupan los componentes gráficos que ayudan a la conexión con la base de datos para poder navegar entre vistas. Por tal motivo, se decide utilizar este framework.

1.2.4.2 Primefaces

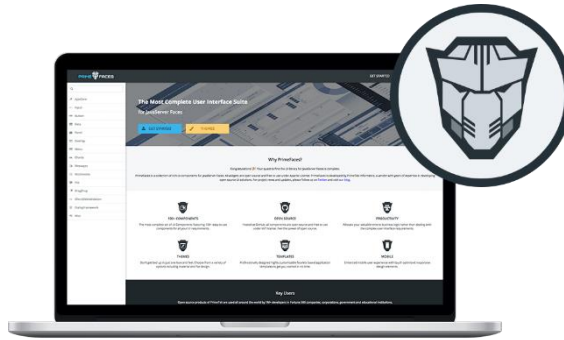


Figura 7-1: Primefaces para JSF

Fuente: (PrimeTek, 2009)

Definición

Es una librería de componentes visuales de código abierto para Java Server Faces, es fácil de instalar, dado que no es pesado; siendo uno de los framework más populares del mundo, debido a que no depende de muchas librerías y no requiere configuraciones para utilizarlo. Además, está bajo la licencia de Apache License V2 y permite la integración con otros componentes como RichcFaces. (Viñé Lerma, 2010)

Primefaces es una librería que ofrece varios componentes que ayudan y facilitan la creación de aplicaciones web; se divide en tres módulos: Set para interfaces de usuario, set para crear managed beans (utiliza un módulo denominado Optimus y Guice para simplificar la navegabilidad) y un tercer módulo denominado FacesTrace (facilita el monitoreo de aplicaciones JSF); cada uno de estos módulos es independiente, lo que permite crear cualquier tipo de combinación de Framework. (Cordova Jaramillo y Villamagua Morales, 2015, p. 19)

Ajax (Java script y XML Asíncrono), son tecnologías muy independientes que se unen de una manera sorprendente, creando una tecnología nueva. Ajax está formada por HTML y CSS para la creación de presentaciones estandarizadas DOM (Modelo de Objetos del Documento). (Cordova Jaramillo y Villamagua Morales, 2015, p. 19)

Características de Primefaces

- Tiene soporte nativo de Ajax que incluye Push/Comet.
- Cuenta con un kit para crear aplicaciones web y aplicaciones móviles.
- Es compatible con otras librerías de componentes como: JBoss RichFaces.
- Usa java script no intrusivo (no aparece en línea dentro de los elementos, sino dentro de un bloque <script>).

- Las librerías en las que se basa utilizan el soporte de jQuery y jQuery UI para los efectos visuales.
- Incorpora el concepto de skins, utilizando ThemeRoller, y dispone de 26 temas prediseñados.
- Es un proyecto open source (código abierto) activo y bastante estable entre versiones. (Viñé Lerma, 2010)

Desventajas

- Cuando se utiliza el soporte de Ajax se debe indicar explícitamente por medio de atributos específicos cada uno de los componentes.
- No son compatibles completamente entre versiones.
- No se puede utilizar el soporte de Ajax de JSF2 con los componentes de Primefaces. (Viñé Lerma, 2010)

Primefaces es creado y conservado por “Premier Teknoloji”, una empresa desarrolladora de software en Turquía especializada en Java EE. Además, es distribuido como un único archivo .jar que puede ser descargado junto con su documentación. Primefaces cuenta con amplias extensiones como: (Rios Luna, 2014, p. 29)

- Conjunto de componentes como: HTML Editor, Dialog, Auto Complete, Charts, etc.
- Es ligero sin configuraciones ni dependencias y se distribuye en un único archivo jar.
- También incluye componentes para dispositivos móviles.
- Tiene más de 35 temas pre construidos.
- Cuenta con herramientas de interfaz de usuario.
- Y está ampliamente documentado. (Rios Luna, 2014, p. 29)

Componentes básicos de Primefaces

- CommandButton.
- InputText/ InputTextArea.
- Calendar/Growl.
- CommandButton y CommandLink.
- DataTable.
- Drag and Drop.
- Form.

- InputText.
- InputTextArea.
- Componentes SelectOne.
- Mensajes.
- SelectMany.
- SelectItem.
- OutputText.
- PanelGrid. (Cordova Jaramillo y Villamagua Morales, 2015, p. 20)

Componentes de selección que utiliza Primefaces

- Boolean.
- Simple.
- Múltiple. (Cordova Jaramillo y Villamagua Morales, 2015, p. 21)

Se selecciona Framework Primefaces para desarrollar el sistema, dado que es compatible con el IDE Netbeans, además usa java script no intrusivo y es estable entre versiones.

1.2.4.3 Bootstrap



Figura 8-1: Dispositivos compatibles con Bootstrap

Fuente:(Hernandez, 2016)

Definición

Es un framework que facilita el desarrollo de la interfaz web, está basado en los estándares de HTML, CSS y Java script; tiene incorporado varias plantillas prediseñadas, formularios, botones y menús, así como también, otros componentes que facilitan y agilizan el desarrollo de la interfaz de las aplicaciones web, para que éstas se adapten de manera dinámica al tamaño del dispositivo del cual el usuario ingresa. (Tituaña Maldonado, 2017, p. 15)

Este framework fue desarrollado por Twitter, el cual trabaja con JavaScript y CSS; éstos ayudan a crear interfaces adaptables al tamaño del dispositivo que se utiliza; es decir, que la página creada se vea de una forma correcta, tanto en una PC, en una Tablet, en un móvil o en cualquier dispositivo que tenga un navegador web, la técnica de desarrollo se llama “responsive desing”. (Jiménez Chilán, 2017, pp. 13–14)

Bootstrap se genera a partir del módulo de CSS3 que utiliza jQuery; con la cual se trabaja el tamaño de las pantallas, tanto en el porcentaje como en sus dimensiones para tener fluidez en los navegadores web, esto ayuda al menor consumo de recursos en cada uno de los dispositivos, haciendo más rápida la transferencia de los datos. (Jiménez Chilán, 2017, p. 14)

Características de Bootstrap

- Permite dimensionar de una forma automática el tamaño y la vista.
- Tiene CSS (Integración de hojas de estilo en cascada), y LESS (Lenguaje de hojas de Estilo Dinámico), incorporados para crear una interfaz dinámica que se ajuste automáticamente a los navegadores web, permitiendo mostrar estilos elegantes y estéticos.
- Cuenta con una librería JavaScript, con el fin de brindar elementos adicionales que utilicen componentes ágiles y eficientes.
- Utiliza el método denominado Sistema grid o de rejillas, el cual hace que se divida en dos segmentos de 60 pixeles, con un espacio de segregación de 20 pixeles. Se le conoce también como mobile first grid para dispositivos móviles.
- Contiene gran cantidad de documentación en línea, tanto en libros digitales como en blogs de internet; donde se detalla paso a paso el funcionamiento de cada uno de sus componentes, y de cómo usarlos de forma más simple. (Tituaña Maldonado, 2017, pp. 15-6–17)

Estructura de Bootstrap

Cuenta con tres carpetas que son: La carpeta CSS con archivos de hojas de estilo, la carpeta JS donde se ubican los archivos java script y por último la carpeta para las imágenes. (Tituaña Maldonado, 2017, p. 19)

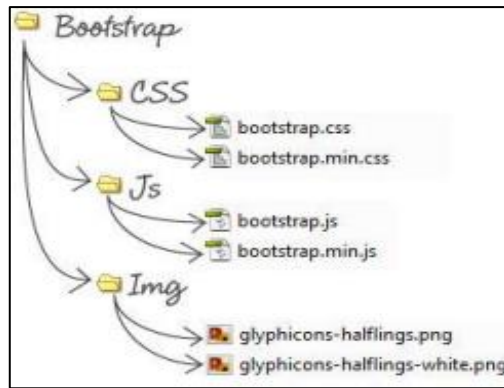


Figura 9-1: Estructura de Bootstrap

Fuente: (Tituaña Maldonado, 2017, p. 19)

Arquitectura Bootstrap

La arquitectura de Bootstrap está basada principalmente en hojas de estilos CSS y en lenguaje de hojas dinámico (LESS), además tiene incorporado plugins de java script, para que funcionen correctamente necesitan la librería jQuery. (Tituaña Maldonado, 2017, p. 19)

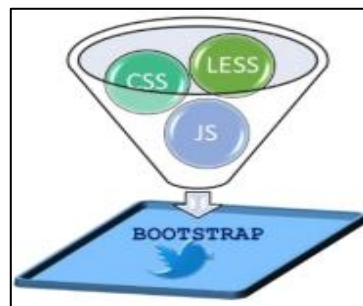


Figura 10-1: Estructura de Bootstrap

Fuente: (Tituaña Maldonado, 2017, p. 19)

Lenguajes programación que se integran con Bootstrap

- Active Server Pages (ASP).
- Delphi.
- Hypertext Pre-Processosr (PHP).
- Java Server Faces (JSF).
- Java Serve Pages (JSP).
- Python.
- Ruby. (Tituaña Maldonado, 2017, p. 20)

Componentes de Bootstrap

- Glyphicons.
- Dropdowns.
- Button groups.
- Button dropdowns.
- Input groups.
- Navs.
- Navbar.
- Breadcrumbs.
- Pagination.
- Labels.
- Badges.
- Jumbotron.
- Page header.
- Thumbnails.
- Alerts.
- Progress bars.
- Media object.
- List group.
- Panels.
- Wells. (Tituaña Maldonado, 2017, pp. 21–22)

Se selecciona Bootstrap para crear interfaces con CSS y java script, así como para adaptar el tamaño de la pantalla del dispositivo donde se navegará.

1.2.5 Sistema Gestor de Base de Datos (SGBD)

La gestión de bases de datos ha ido evolucionando desde una aplicación informática especializada hasta una parte esencial de un entorno informático moderno, por tal motivo se han convertido en una parte primordial para la enseñanza de la informática. (Silberschatz et al., 2002)

Definición

Un sistema gestor de base de datos, es un software o un conjunto de programas, que permite crear y mantener una base de datos. Actúa como interfaz entre los programas de la aplicación, usuarios

y el sistema operativo. Además, proporciona un entorno eficiente a la hora de almacenar y recuperar información de la base de datos. (Cobo Yera, 2009, p. 7)

El objetivo principal de un sistema gestor de base de datos es: facilitar una manera de almacenar y recuperar la información de una base de datos (BD) de manera práctica y eficiente. Estos sistemas gestores se diseñan principalmente para gestionar cantidades grandes de información. La gestión de datos involucra la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la información. (Silberschatz et al., 2002, p. 1)

La característica principal de un sistema gestor de base de datos es: que debe proporcionar la fiabilidad de toda la información almacenada, a pesar de que existan caídas del sistema o intentos de acceso sin la debida autorización. Si aquellos datos van a ser compartidos entre varios usuarios, el sistema debe evitar posibles resultados anómalos. (Silberschatz et al., 2002, p. 1)

Un sistema de base de datos (SBD), es una colección de archivos interrelacionados que permiten a los usuarios acceder y modificar estos archivos. Además, el propósito principal de un SBD es proporcionar a los usuarios una visión más abstracta de los datos. Es decir, el sistema esconde algunos detalles de cómo se almacenan y mantienen los datos. (Silberschatz et al., 2002, p. 3)

Aplicaciones de los sistemas de base de datos (SBD)

A continuación, algunos de los lugares donde se aplican los sistemas:

- Líneas Aéreas.
- Banca.
- Universidades.
- Transacciones de tarjeta de crédito.
- Finanzas.
- Telecomunicaciones.
- Recursos humanos.
- Producción.
- Ventas. (Silberschatz et al., 2002, p. 1)

La importancia de los sistemas de bases de datos (SBD) se la puede juzgar de otra manera, dado que, actualmente los vendedores de sistemas de bases de datos como: Oracle están entre las mayores compañías software en el mundo, y los SBD forman una parte muy importante de la

línea de productos de compañías más diversificadas como Microsoft e IBM. (Silberschatz et al., 2002, p. 2)

Se selecciona un sistema gestor de base de datos, dado que, permite a los usuarios definir, crear y mantener la base de datos proporcionando acceso a la misma, además, interactúa con los usuarios de la aplicación y la base de datos, permitiendo la inserción, actualización, eliminación y consulta de información mediante el manejo de los datos.

1.2.5.1 PostgreSQL 9.4



Figura 11-1: Logo de PostgreSQL

Fuente: (López, 2017)

Definición

PostgreSQL, es un motor de bases de datos (BD) relacional, el cual tiene características como: integración de datos, confiabilidad y desempeño, al ser una base de datos libre de licencia o de código abierto puede ser usado y modificado a conveniencia del usuario y se lo puede usar en diferentes sistemas operativos. (Kasián y Reyes, 2012)

En la actualidad, PostgreSQL está presente en distintas áreas como gestor de datos, tales como en la industria farmacológica, educación, el comercio, en juegos de azar, proyectos de código abierto, salud, manufactura, tecnología y telecomunicaciones. Ya que puede ser ejecutado en los principales sistemas operativos como: Linux, UNIX AIX, BSD, HP-UX, SGIIRIX, MacOSX, Solaris, Tru64 y Windows. Este soporta textos, imágenes, sonidos y video e incluye interfaces para C, C++, Java, Perl, Python, Ruby, Tcl y Open Data base Connectivity ODBC. (López, 2017)

*PostgreSQL es **Full ACID** complaciente (Atomicity, Consistency, Isolation and Durability)*

- *Atomicidad (Indivisible)*, es la propiedad que asegura que la operación se ha realizado o no, y por lo tanto ante una falla del sistema no puede quedar a medias.

- *Consistencia*, es la propiedad que asegura que sólo se empieza aquello que se puede acabar. Por lo tanto, se ejecutan aquellas operaciones que no van a romper la reglas y directrices de integridad de la base de datos.
- *Aislamiento*, es la propiedad que asegura que una operación no puede afectar a otras. Esto asegura que dos transacciones sobre la misma información nunca generarán ningún tipo de error.
- *Durabilidad*, es la propiedad que asegura que, una vez realizada la operación, ésta persistirá y no se podrá deshacer, aunque falle el sistema. (Aliaga y Miani, 2007)

Además, tiene más características como:

- Correr en casi todos los sistemas operativos: Linux, Unix, BSDs, Mac OS, BIOS, Windows, etc.
- La documentación muy bien organizada, pública y libre, con comentarios de los usuarios.
- Comunidades en castellano bastantes activas.
- Bajo Costo de Total Propiedad (TCO) y rápido Retorno de la Inversión Inicial (ROI).
- Es altamente adaptable a las necesidades del cliente. (Aliaga y Miani, 2007)

Ventajas

- *Posee una gran escalabilidad*. - Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una cantidad mayor de peticiones simultáneas de manera apropiada.
- *Implementa el uso de rollback's, subconsultas y transacciones*. - Hace que su funcionamiento sea más eficaz, para así ofrecer soluciones en campos en las que MySQL no podría.
- Tiene la capacidad de comprobar la integridad referencial y la de almacenar procedimientos en la propia base de datos, comparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle. (Valdiviezo Serrano y Guacho Minta, 2012, p. 71)
- Posee un código abierto.
- Es flexible y estable.
- Es multiplataforma.
- Tiene una gran capacidad de almacenamiento.
- Posee un alto rendimiento. (Toapanta, 2018, p. 21)

Desventajas

- Consume gran cantidad de los recursos hardware.
- La Sintaxis de comandos no es intuitivo.
- Es de dos a tres veces más lento que MySQL.
- Tiene un límite de 8K por fila, aunque se puede aumentar a 32K, con una disminución considerable del rendimiento. (Valdiviezo Serrano y Guacho Minta, 2012, p. 72)

PostgreSql soporta una gran parte del estándar SQL y ofrece muchas características modernas como:

- Consultas(queries) complejas.
- Subconsultas(Sub-queries).
- Foreign keys (claves secundarias).
- Esquemas.
- Triggers(desencadenadores).
- Vistas.
- Transacciones.
- MVCC (Multiversion Concurrency Control).
- Streaming Replication (replica de transmisión).
- Hot Standby. (López, 2017)

Una de las funcionalidades que dispone PostgreSql son los llamados disparadores o triggers. Un disparador, es una función que se ejecuta automáticamente al realizar algún *INSERT*, *UPDATE* o *DELETE* en una tabla específica. (Jiménez Chilán, 2017, p. 13)

A un disparador se puede definir para que:

- Ocurra ANTES de cualquier *INSERT*, *UPDATE* ó *DELETE*.
- Ocurra DESPUÉS de cualquier *INSERT*, *UPDATE* ó *DELETE*.
- Se ejecute por cada línea afectada por un comando SQL (row-level trigger).
- Se ejecute una sola vez por comando SQL (statement-level trigger). (Jiménez Chilán, 2017, p. 13)

Sin duda alguna se escoge PostgreSql debido a que, es un servidor de base de datos muy conocido y utilizado. Además, es multiplataforma, de fácil manejo, tiene estabilidad y seguridad en cuanto a la información que se almacenará.

1.2.6 Metodología de desarrollo

Dentro del desarrollo de software y la gran necesidad que los proyectos lleguen al éxito para obtener productos de gran valor para los usuarios, se generan grandes cambios en las metodologías adoptadas por los equipos, para cumplir los objetivos, puesto que, algunas tecnologías se adaptan mejor que otras al desarrollo del proyecto brindando mejores ventajas. Por tal motivo, la importancia de una metodología robusta debe ser ajustada a un equipo para cumplir con las metas y satisfacer las necesidades que se definen desde el inicio del proyecto. (Figuerola et al., 2007, p. 1)

Las metodologías tradicionales siempre imponen una disciplina de trabajo sobre el proceso de desarrollo del software, con la finalidad de conseguir un software eficiente. Para ello, se hace un énfasis en la planificación total de todo el trabajo a realizarse, una vez que está todo detallado se comienza el ciclo de desarrollo del proyecto software. Parcialmente, se debe centrar en el control del proceso, mediante una dura definición de roles, actividades, artefactos, herramientas y documentación detallada. (Quingatuña Moreano y Nogales Guerrero, 2016, p. 30)

En cambio, las metodologías ágiles nacen con una respuesta a los problemas que se presentan en el desarrollo del proyecto, por tal motivo, se basa en dos aspectos importantes como: retrasar las decisiones y la planificación adaptativa, para permitir una gran potencia en el desarrollo del proyecto a gran escala. (Figuerola et al., 2007, p. 1)

En la actualidad varias metodologías apoyan en el desarrollo de las aplicaciones software como: XP, Scrum, MSF, ASD, DSDM, Crystal Clear, MÉTRICA y RUP, estas metodologías se las diferencia por: los partidarios de las metodologías ‘tradicionales’ y por los partidarios de las metodologías ‘ágiles’ o ‘ligeras’. (Tisalema Paninboza y Muñoz Arias, 2012, p. 115)

1.2.6.1 Metodología SCRUM

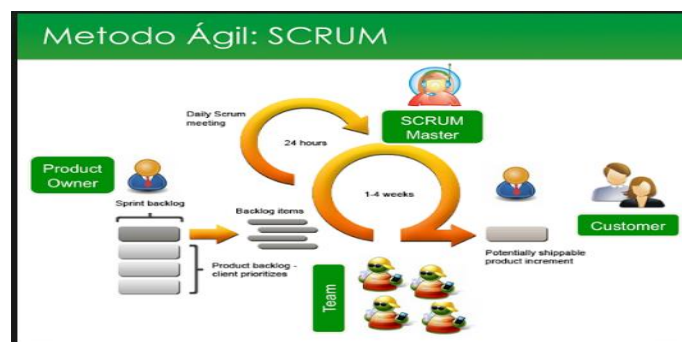


Figura 12-1: Funcionamiento de la metodología Scrum

Fuente: (Toapanta, 2018, p. 22)

La metodología Scrum, es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente en equipo y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan de unas a otras y su clasificación tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. (Acosta Fiallos y Monar Borja, 2016, p. 13)

Scrum, es un proceso ágil y liviano que se lo utiliza para administrar y controlar el desarrollo del software, el desarrollo se realiza en forma iterativa. Cada ciclo o iteración termina con una pieza de software ejecutable que incorpora una nueva funcionalidad, las iteraciones en general tienen una duración entre 2 y 4 semanas. Scrum, se enfoca en priorizar el trabajo en función del valor que tenga para el negocio, maximizando la utilidad de lo que se construye y el retorno de inversión, está diseñado principalmente para adaptarse a cambios en cuanto a los requerimientos. Los requerimientos se revisan y ajustan durante el proyecto en intervalos ya sean cortos y regulares, de esta forma se le puede adaptar en tiempo real al producto que se está construyendo de acuerdo a las necesidades del cliente. (Figuerola et al., 2007, p. 7)

Scrum se basa en tres pilares fundamentales:

- *Transparencia.* - Los aspectos del proceso deben ser visibles para los responsables del resultado del producto
- *Inspección.* - Se inspecciona consecutivamente los artefactos y el progreso del proyecto, para llegar hacia un objetivo y detectar las dificultades previstas en el mismo.
- *Adaptación.* - Se debe ajustar el producto cuando se detecte alguna variación considerable, esto se debe hacer lo más pronto posible para minimizar una desviación adicional. (Schwaber y Sutherland, 2017)

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto, especialmente indicado entornos complejos, donde se necesita obtener rápidamente resultados. Además, los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales. Se utiliza para resolver situaciones en que no se está entregando al cliente lo que realmente necesita, cuando las entregas se alargan demasiado, los costos se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente en el desarrollo de producto. (Acosta Fiallos y Monar Borja, 2016, p. 14)

Fases de SCRUM

- *Gestión de la Lista del Producto (Product Backlog)*
 - Debe expresar claramente los elementos del Product Backlog.
 - Gestiona la lista de requerimientos sobre las funcionalidades del producto.
 - Ordena los elementos en el Product Backlog para alcanzar los objetivos y misiones.
 - Optimiza el valor del trabajo que el equipo de desarrollo ejecuta.
 - Asegura que el Product Backlog sea visible, transparente y claro para todos los del equipo.
 - Asegura que el equipo de desarrollo entienda los elementos del Product Backlog al nivel necesario.
 - Es elaborado por el Product Owner y las funciones están priorizadas según la importancia para el proyecto. (Mejia Broncano, 2016, p. 17)
- *Lista de Pendientes del Sprint (Sprint Backlog)*
 - Es un bloque de tiempo de un mes o menos durante el cual se crea un incremento de producto “Terminado” utilizable y potencialmente desplegable.
 - Es un subconjunto de ítems del Product Backlog, que son seleccionados por el equipo para realizar durante el Sprint.
 - El equipo establece la duración de cada Sprint. Cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint anterior. (Schwaber y Sutherland, 2017)
- *Reunión de planificación del sprint (Sprint Planning Meeting)*
 - Reunión al comienzo de cada SPRINT, se define cómo se va a enfocar el proyecto que viene del Product Backlog en cuanto a las etapas y los plazos.
 - Cada Sprint está compuesto por diferentes actividades a realizarse. (Quingatuña Moreano y Nogales Guerrero, 2016, p. 32)
- *Scrum diario (Daily SCRUM)*
 - Es una reunión temporal que se realiza a diario mientras dura el periodo del sprint.
 - El Scrum Diario se realiza a la misma hora y en el mismo lugar todos los días para reducir la complejidad.
 - Se responden individualmente a tres preguntas: ¿Qué hice ayer?, ¿Qué voy a hacer hoy?, ¿Qué ayuda necesito?
 - El SCRUM Master debe solucionar los problemas o dificultades que se presentan. (Quingatuña Moreano y Nogales Guerrero, 2016, p. 32)
- *Revisión de Sprint (Sprint Review)*
 - Al final del sprint se realiza una Revisión de Sprint para inspeccionar el incremento y adaptar el Product Backlog si es necesario.

- Durante la Revisión de Sprint el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint.
- Se revisa el sprint terminado, debe haber un avance claro y tangible para presentárselo al cliente. (Mejia Broncano, 2016, p. 19)
- *Retrospectiva de Sprint (Sprint Retrospective)*
 - Inspeccionar cómo fue el último Sprint en cuanto a personas, relaciones, procesos y herramientas
 - Se debe identificar y ordenar los elementos importantes que salieron bien y las posibles mejoras, para no volver a repetir los errores.
 - Se crea un plan para implementar mejoras con el equipo Scrum que desempeña su trabajo. (Mejia Broncano, 2016, p. 18)

El Equipo Scrum (Scrum Team)

Los Equipos Scrum son multifuncionales y auto organizados. Los equipos multifuncionales tienen todas las competencias necesarias para llevar a cabo el trabajo sin depender de otras personas que no son parte del equipo, en cambio los equipos auto organizados eligen la mejor forma de llevar a cabo su trabajo y no son dirigidos por personas externas al equipo. El modelo de equipo en Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad. Por tal motivo, los Equipos Scrum entregan productos de forma iterativa e incremental, maximizando las oportunidades de obtener retroalimentación, haciendo las entregas incrementales de producto “Terminado” asegurando que siempre esté disponible una versión potencialmente útil y funcional del producto. (Schwaber y Sutherland, 2017, p. 5)



Figura 13-1: Ciclo de desarrollo Scrum

Fuente: (Veloz, 2016, p. 61)

- *Product Owner (Dueño del Producto)*
 - Es la única persona responsable de gestionar el Product Backlog
 - Responsable de maximizar el valor del producto y el trabajo del Equipo de Desarrollo.
 - Representa los deseos de un comité en el Product Backlog
 - Las decisiones del *Product Owner* se reflejan en el contenido y en la priorización del Product Backlog. (Schwaber y Sutherland, 2017, p. 5)
- *Scrum Master*
 - Es el encargado del seguimiento de la teoría por parte del equipo.
 - Es el encargado de garantizar que el modelo y la metodología funcionen perfectamente
 - Trabaja con el Product owner para la construcción, diseño y conservación del product backlog,
 - Se encarga de dar solución a los inconvenientes que se presenten. (Aucancela Quinteros y Cajilema Quishpi, 2017, p. 32)
- *Equipo de desarrollo Scrum (Scrum Development Team)*
 - Son los profesionales que realizan el trabajo de entregar un incremento del producto “Terminado” con el potencial de ser puesto en producción al final del Sprint
 - Son los encargados de desarrollar y cumplir lo que asigna el Product Owner.
 - Un equipo puede estar conformado entre tres y nueve miembros, pudiendo ser esta cantidad elegida en dependencia del contexto en que se va a realizar el trabajo. (Mejia Broncano, 2016, p. 18)
- *Cliente*
 - Es el que recibe el producto final y puede influenciar en el proceso, entregando sus ideas o comentarios respecto al desarrollo. (Quingatuña Moreano y Nogales Guerrero, 2016, p. 32)

Se selecciona la metodología Scrum dado que ayudara en el desarrollo del sistema, dividiendo los actores y las acciones a realizarse acorde a los requerimientos del usuario para alcanzar con el objetivo deseado. Dando entregas parciales y regulares del producto, desarrollándolo de forma incremental y evolutiva.

1.2.7 Calidad del Software

La Calidad se determina en función de las características que hacen competitivo un producto al satisfacer las necesidades ya sean explícitas e implícitas de los usuarios finales. Es decir, el cliente

tiene la responsabilidad de suministrar las especificaciones correctas para el desarrollo del proyecto. En particularidad, se define la calidad como: satisfacción de las necesidades y cumplimiento de las especificaciones para cumplir con el propósito del cliente. Por otra parte, los requisitos de calidad hacen referencia al conjunto de requisitos expresados en forma cuantitativa o cualitativa respecto a las características de los productos. Para hacer posible su realización y examen; los requisitos pueden ser a conveniencia y basados a partir de las necesidades del mercado interno de una organización o de la sociedad en sí, esto incluye lógicamente requisitos jurisdiccionales y reglamentarios. (Ruiz et al., 2006)

1.2.7.1 Norma ISO/IEC 9126

En 1985 comenzó el desarrollo de la norma ISO/IEC 9126 y como primer paso, el comité técnico intentó organizar las propiedades de la calidad de forma sistemática, pero fue un fracaso rotundo debido a que los expertos las interpretaban de manera diferente. A continuación, los requisitos para la elección de las características descritas en la norma ISO/IEC 9126, desarrolladas por el comité técnico. (Ortega Cabrera, 2010, pp. 11–12)

- Resguardar todos los aspectos de la calidad del software resultante de la definición de calidad ISO.
- Detallar la calidad del producto con un mínimo de sobre posición.
- Estar lo más cerca posible a la terminología ya establecida.
- Formar un conjunto de no más de seis a ocho características por razones de claridad y manejo.
- Identificar las áreas de los atributos de los productos de software para un mayor refinamiento. (Ortega Cabrera, 2010, pp. 11–12)

La norma ISO (Organización Internacional para la Estandarización) /IEC (Comisión Electrotécnica Internacional) es un estándar internacional para la evaluación de calidad de productos de software. Esta norma está dividida en cuatro partes: Modelo de Calidad, Métricas Externas, Métricas Internas y Calidad en uso. (Ortega Cabrera, 2010, p. 10)

La Norma ISO/IEC 9126 categoriza los atributos de calidad del software en seis características: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad. (Ruiz et al., 2006)

Características de la calidad del software

- *Funcionalidad.* – Es la capacidad del software para proporcionar funciones que satisfagan las necesidades explícitas e implícitas cuando el software se utiliza en las condiciones detalladas.
- *Fiabilidad.* - Es la capacidad del software para mantener un nivel específico de rendimiento cuando se utiliza en las condiciones dadas.
- *Usabilidad.* - Es la capacidad del software de facilitar al usuario la interacción y el aprendizaje de una manera atractiva, depende del tipo de "uso" que se espera y tipo de "usuario" que utilizará el producto
- *Eficiencia.* - Es la capacidad del software para proporcionar el rendimiento apropiado, relativo a la cantidad de los recursos utilizados, bajo las especificaciones determinadas.
- *Mantenibilidad.* - Es la capacidad del software para ser modificado, las modificaciones pueden incluir correcciones, mejoras o adaptación a cambios en el entorno, en los requisitos o en las especificaciones funcionales.
- *Portabilidad.* - Es la capacidad del software de ser transferido de un entorno hacia a otro. (Quingatuña Moreano y Nogales Guerrero, 2016, p. 33)

La Norma ISO/IEC 9126 fue seleccionada porque contiene 2 características importantes para medir la calidad del sistema como son la usabilidad y la eficiencia, que son atributos muy importantes

Métricas de usabilidad

Usabilidad. – Es la capacidad del producto de software de ser comprendido, aprendido, usado y atractivo para el usuario, cuando este es usado en condiciones específicas. Además, este está compuesto por subcaracterísticas como: comprensibilidad, facilidad de aprendizaje, operabilidad, atractivo y cumplimiento de usabilidad. (Ortega Cabrera, 2010, pp. 13–14)

La usabilidad mide la facilidad de uso del sitio, de cómo se siente el usuario navegando por él y las facilidades que el sitio provee para que el usuario se sienta cómodo navegando por él. (Pulido Granados y Medina García, 2008)

Subcaracterísticas de la Usabilidad

- *Comprensibilidad.* - Es la capacidad del producto software para permitir al usuario poder entender si es adecuado y como se puede utilizar para determinadas tareas y condiciones.
- *Facilidad de aprendizaje.* - Es la capacidad del producto software para permitir al usuario aprender de su aplicación.
- *Operabilidad.* - Es la capacidad del producto software para permitir al usuario que lo opere y controle.
- *Atractivo.* - Es la capacidad del producto software para lograr que sea atractivo hacia el usuario.
- *Cumplimiento de usabilidad.* - Es la capacidad del producto software para que se adhiera a las normas, regulaciones o convenciones en leyes y reglamentos relativos a la usabilidad. (Ortega Cabrera, 2010, pp. 14–15)

Métricas de Eficiencia

Eficiencia: Es la capacidad que tiene un producto de software para proveer el desempeño apropiado relacionado a la cantidad de recursos usados, bajo condiciones determinadas. Esta métrica se usa prácticamente para determinar la capacidad del software al realizar el testeo y la ejecución, las cuales son relativas a la cantidad de recursos usados, bajo ciertas condiciones para medir la eficiencia. (Pilco Guachi, 2018, p. 19)

Subcaracterísticas de la eficiencia

- *Métricas de comportamiento con respecto al tiempo.* – Es la capacidad del producto software para proporcionar los tiempos de repuesta y de proceso al ejecutar sus funciones. (Aucancela Quinteros y Cajilema Quishpi, 2017, p. 35)
- *Utilización de los recursos.* – En este tipo de métricas se reflejan un conjunto de características y atributos que permiten pronosticar la utilización de los recursos hardware donde se esté ejecutando el sistema informático o proyecto. (Aucancela Quinteros y Cajilema Quishpi, 2017, p. 35)

CAPÍTULO II

2 MARCO METODOLÓGICO

En este capítulo se describe el tipo de investigación utilizado para obtener información sobre la venta y crianza de cerdos en la granja, también se describe la población y muestra que ayudará a evaluar la usabilidad y rendimiento del sistema, además se presenta la metodología utilizada para dar seguimiento.

2.1 Tipo de investigación

Para el desarrollo del sistema informático se utiliza el tipo de investigación aplicada, debido a que se basa en conocimientos existentes dirigidos al desarrollo tecnológico, productos o dispositivos y al establecimiento de nuevos procesos y sistemas. Además, utiliza metodologías de desarrollo de software, características y métricas que ayudan a colocarlo en el entorno social.

2.2 Población y muestra

La muestra es el subconjunto de una población existente, mediante la cual se evalúa el sistema informático, la muestra tomada para el estudio del sistema es del 100% de la población.

La aplicación es utilizada por un usuario en el área administrativa y por cinco empleados quienes laboran en la granja porcina “Platanillos” ubicada en la ciudad de Piñas provincia de El Oro.

2.3 Técnicas de investigación

La técnica primordial utilizada para el desarrollo del sistema web para el control de venta y crianza de cerdos es la entrevista y la observación directa, mediante las cuales se obtuvo información relevante sobre los procesos administrativos que se realizan en la granja porcina.

Para la evaluación del sistema informático se optó por una entrevista a los usuarios que van a utilizarlo, esta entrevista ayuda a obtener información muy valiosa y de una forma muy práctica acerca del sistema.

El modelo de la entrevista realizada permite medir los parámetros de usabilidad como son: Comprensibilidad, Facilidad de aprendizaje, Operabilidad y Atractivo, todo esto se aplica a los

usuarios que hayan utilizado el sistema para el control de venta y crianza de cerdos. De esta manera se obtiene estadísticas y mediante la tabulación de dichas entrevistas se muestra los resultados, los cuales evalúan cada uno de los parámetros ya mencionados anteriormente. En el **ANEXO E** se presenta el modelo aplicado a los usuarios que utilizan el sistema.

En cuanto a la medición de lo que es el rendimiento se toma en cuenta todo lo relativo al tiempo, considerándolo como instrumento ideal para medir el tiempo que el personal que administra el sistema se tarda en hacer cada actividad tanto manual como usando el sistema. En el **ANEXO F** se presenta el modelo de medición del tiempo utilizado.

2.4 Metodología Scrum para el desarrollo del sistema

Se decidió escoger la metodología Scrum para el desarrollo del sistema porque tiene las características de ser ágil, flexible y permitir que el cliente forme parte del equipo de trabajo, esto hace que el producto sea entregado de una forma muy satisfactoria al cliente.

La metodología ágil tiene tres fases que son realmente principales para la evolución del sistema, las cuales son: La Fase de planificación, la fase de desarrollo y la fase de finalización.

2.4.1 Fase de planificación

En esta fase se realiza la calendarización del proyecto con el fin de que exista coordinación entre los miembros del equipo donde se elaboran los llamados Sprint o iteraciones de acuerdo a la prioridad que tenga cada requerimiento.

Cabe mencionar que cada sprint se compone de 80 horas laborables, lo cual se estableció con una duración de 105 días, logrando con un total de 840 puntos estimados.

La planificación se la presenta en un diagrama Gantt planificado y programado en el **ANEXO A**, en el cual se elaboraron los sprints según la prioridad de cada uno de los requerimientos, y metáforas del sistema que no son requeridas por el cliente si no por el desarrollador del sistema, que son de suma importancia para él.

2.4.1.1 Requerimientos

Los requerimientos son las necesidades del cliente para adquirir un sistema 100% funcional, estableciendo que el mismo tenga procesos, consultas, reportes, interfaces amigables, seguridad y elementos necesarios para alcanzar una dimensión real del proyecto.

Los requerimientos se establecieron mediante reuniones con el administrador y los empleados de la granja, para esto se utiliza la metodología ágil Scrum, la cual emplea el llamado Product Backlog para especificar la lista de requerimientos técnicos y los de usuario de acuerdo a la prioridad que se le haya asignado. Además, para precisar el esfuerzo que se necesita en el desarrollo de cada una de las historias ya sean técnicas o de usuario, se utiliza el método denominado T- shirt que emplea las tallas de camiseta para realizar la estimación de las actividades del proyecto.

Tabla 1-2: Método de estimación T-Shirt

Talla	Puntos Estimados	Horas De Trabajo
XXS	8	8
XS	16	16
S	24	24
M	40	40
L	80	80

Realizado por: Karen Pardo. 2019

En las **Tablas 2-2** y **3-2** se presenta el product backlog de las historias de usuario y las técnicas, detallando los requerimientos con su respectiva prioridad y puntos estimados.

Tabla 2-2: Product Backlog - Historias de Usuario

Historias de Usuario			
Id	Requerimientos	Prioridad	Puntos Estimados
HU_01	Ingreso de usuario	Alta	16
HU_02	Modificación de usuario	Media	16
HU_03	Eliminación de usuario	Baja	8
HU_04	Listado de usuarios	Media	16
HU_05	Ingreso del tipo de producto	Alta	8
HU_06	Modificación del tipo de producto	Media	8
HU_07	Eliminación del tipo de producto	Baja	8
HU_08	Listado del tipo de productos	Media	8
HU_09	Ingreso de unidad de medida	Alta	8

HU_10	Modificación de unidad de medida	Media	16
HU_11	Eliminación de unidad de medida	Baja	8
HU_12	Listado de unidad de medida	Media	8
HU_13	Ingreso de productos	Alta	16
HU_14	Modificación de productos	Media	16
HU_15	Eliminación de productos	Baja	8
HU_16	Listado de productos	Media	16
HU_17	Ingreso del tipo de sala	Alta	8
HU_18	Modificación del tipo de sala	Media	8
HU_19	Eliminación del tipo de sala	Baja	8
HU_20	Listado del tipo de sala	Media	8
HU_21	Ingreso de sala	Alta	16
HU_22	Modificación de sala	Media	16
HU_23	Eliminación de sala	Baja	8
HU_24	Listado de salas	Media	16
HU_25	Ingreso del tipo animal	Alta	8
HU_26	Modificación del tipo animal	Media	8
HU_27	Eliminación del tipo animal	Baja	8
HU_28	Listado del tipo animal	Media	8
HU_29	Registro de animales en una sala	Alta	16
HU_30	Modificación de animal	Media	16
HU_31	Eliminación de animal	Baja	8
HU_32	Listado de animales por sala	Media	16
HU_33	Listado de animales por salas caducadas	Media	16
HU_34	Asignación de animales a una nueva sala	Alta	24
HU_35	Venta de animales	Alta	24
HU_36	Listado de animales a vender	Media	16
HU_37	Compra de productos	Alta	16
HU_38	Listado de productos a comprar	Media	16
HU_39	Salida de productos	Media	16
HU_40	Listado de productos consumidos	Media	8
HU_41	Ingreso de proveedores	Alta	16
HU_42	Modificación de proveedores	Media	16
HU_43	Eliminación de proveedores	Baja	8
HU_44	Listado de proveedores	Media	16
HU_45	Reporte de animales por sala	Media	8
HU_46	Reporte de productos alimenticios	Media	8
HU_47	Reporte de productos medicinales	Media	8
HU_48	Reporte de las salas dada una fecha	Media	24
HU_49	Reporte de ventas por fecha	Media	16
HU_50	Reporte de animales en venta	Media	16

Realizado por: Karen Pardo. 2019

Tabla 3-2: Product Backlog - Historias Técnicas

Historias Técnicas			
Id	Requerimientos	Prioridad	Puntos Estimados
HT_01	Estudio de las herramientas	Alta	24
HT_02	Análisis y gestión de riesgos	Media	16
HT_03	Definición del estándar de codificación	Media	24
HT_04	Diseño de la base de datos	Alto	40
HT_05	Diseño de la arquitectura del sistema	Media	24

HT_06	Diseño de las interfaces de usuario	Media	40
HT_07	Desarrollo del manual de usuario	Media	40

Realizado por: Karen Pardo. 2019

2.4.1.2 Personas y roles del proyecto

En la **Tabla 4-2** se detalla el personal con sus respectivos roles que contaron para el desarrollo del sistema.

Tabla 4-2: Personas y roles

Entidad /Persona	Rol
Granja porcina "Platanillos"	Product Owner
Dr. Julio Santillán	Scrum Master
Sr. Vicente Pardo	Stake Holder
Srta. Karen Pardo	Developer

Realizado por: Karen Pardo. 2019

2.4.1.3 Tipos y roles de usuario

Para el desarrollo del sistema informático se plantean dos tipos de usuario; el administrador de la granja y los empleados, esto se define conjuntamente con el cliente.

Tabla 5-2: Tipos y roles de usuario

Tipos de Usuario	Rol
Administrador	Gestión de usuarios Gestión de proveedores Gestión de productos alimenticios Gestión de medicamentos Gestión de salas Gestión de ventas Gestión de crianza Obtención de reportes
Empleados	Gestión de proveedores Gestión de productos alimenticios Gestión de medicamentos Gestión de salas Gestión de ventas Gestión de crianza Obtención de reportes

Realizado por: Karen Pardo. 2019

2.4.1.4 Plan de entrega

Presentados y recolectados todos los requerimientos del cliente en el product backlog, por consiguiente, se realiza un sprint backlog en el cual se planifican los entregables del proyecto priorizando los requerimientos según la importancia y el costo de los mismos.

Se definen dos semanas de trabajo en cada sprint con una duración de ocho horas diarias, las cuales suman un total de 80 horas por cada sprint.

En la **Tabla 6-2** se presenta el Sprint Backlog con sus actividades a realizarse y su correspondiente fecha de inicio y fecha fin.

Tabla 6-2: Sprint Backlog

Id	Requerimientos	Pts_Est	Total	Fecha Inicio	Fecha Fin
SPRINT 1					
HT_01	Estudio de las herramientas	24	80	lun 02/07/2018	mié 04/07/2018
HT_02	Análisis y gestión de riesgos	16		jue 05/07/2018	vie 06/07/2018
HT_03	Definición del estándar de codificación	24		lun 09/07/2018	mie 11/07/2018
HT_04	Diseño de la base de datos	16		jue 12/07/2018	vie 13/07/2018
SPRINT 2					
HT_04	Diseño de la base de datos	24	80	lun 16/07/2018	mié 18/07/2018
HT_05	Diseño de la arquitectura del sistema	24		jue 19/07/2018	lun 23/07/2018
HT_06	Diseño de las interfaces de usuario	32		mar 24/07/2018	vie 27/07/2018
SPRINT 3					
HT_06	Diseño de las interfaces de usuario	8	80	lun 30/07/2018	lun 30/07/2018
HU_01	Ingreso de usuario	16		mar 31/07/2018	mié 01/08/2018
HU_02	Modificación de usuario	16		jue 02/08/2018	vie 03/08/2018
HU_04	Listado de usuarios	16		lun 06/08/2018	mar 07/08/2018
HU_05	Ingreso del tipo de producto	8		mié 08/08/2018	mié 08/08/2018

HU_06	Modificación del tipo de producto	8		jue 09/08/2018	jue 09/08/2018
HU_09	Ingreso de unidad de medida	8		vie 10/08/2018	vie 10/08/2018
SPRINT 4					
HU_10	Modificación de unidad de medida	16	80	lun 13/08/2018	mar 14/08/2018
HU_08	Listado del tipo de productos	8		mié 15/08/2018	mié 15/08/2018
HU_12	Listado de unidad de medida	8		jue 16/08/2018	jue 16/08/2018
HU_13	Ingreso de productos	16		vie 17/08/2018	lun 20/08/2018
HU_14	Modificación de productos	16		mar 21/08/2018	mié 22/08/2018
HU_17	Ingreso del tipo de sala	8		jue 23/08/2018	jue 23/08/2018
HU_18	Modificación del tipo de sala	8		vie 24/08/2018	vie 24/08/2018
SPRINT 5					
HU_16	Listado de productos	16	80	lun 27/08/2018	mar 28/08/2018
HU_20	Listado del tipo de sala	8		mié 29/08/2018	mié 29/08/2018
HU_21	Ingreso de sala	16		jue 30/08/2018	vie 31/08/2018
HU_22	Modificación de sala	16		lun 03/09/2018	mar 04/09/2018
HU_03	Eliminación de usuario	8		mié 05/09/2018	mié 05/09/2018
HU_07	Eliminación del tipo de producto	8		jue 06/09/2018	jue 06/09/2018
HU_11	Eliminación de unidad de medida	8		vie 07/09/2018	vie 07/09/2018
SPRINT 6					
HU_24	Listado de salas	16	80	lun 10/09/2018	mar 11/09/2018
HU_25	Ingreso del tipo animal	8		mié 12/09/2018	mié 12/09/2018
HU_26	Modificación del tipo animal	8		jue 13/09/2018	jue 13/09/2018
HU_28	Listado del tipo animal	8		vie 14/09/2018	vie 14/09/2018

HU_29	Registro de animales en una sala	16		lun 17/09/2018	mar 18/09/2018
HU_30	Modificación de animal	16		mié 19/09/2018	jue 20/09/2018
HU_15	Eliminación de productos	8		vie 21/09/2018	vie 21/09/2018
SPRINT 7					
HU_32	Listado de animales por sala	16	80	lun 24/09/2018	mar 25/09/2018
HU_33	Listado de animales por salas caducadas	16		mie 26/09/2018	jue 27/09/2018
HU_34	Asignación de animales a una nueva sala	24		vie 28/09/2018	mar 02/10/2018
HU_19	Eliminación del tipo de sala	8		mié 03/10/2018	mié 03/10/2018
HU_23	Eliminación de sala	8		jue 04/10/2018	jue 04/10/2018
HU_27	Eliminación del tipo animal	8		vie 05/10/2018	vie 05/10/2018
SPRINT 8					
HU_35	Venta de animales	24	80	lun 08/10/2018	mié 10/10/2018
HU_36	Listado de animales a vender	16		jue 11/10/2018	vie 12/10/2018
HU_37	Compra de productos	16		lun 15/10/2018	mar 16/10/2018
HU_38	Listado de productos a comprar	16		mié 17/10/2018	jue 18/10/2018
HU_31	Eliminación del animal	8		vie 19/10/2018	vie 19/10/2018
SPRINT 9					
HU_39	Salida de productos	16	80	lun 22/10/2018	mar 23/10/2018
HU_40	Listado de productos a consumir	8		mié 24/10/2018	mié 24/10/2018
HU_41	Ingreso de proveedores	16		jue 25/10/2018	vie 26/10/2018
HU_42	Modificación de proveedores	16		lun 29/10/2018	mar 30/10/2018
HU_43	Eliminación de proveedores	8		mié 31/10/2018	mié 31/10/2018
HU_44	Listado de proveedores	16		jue 01/11/2018	vie 02/11/2018
SPRINT 10					

HU_45	Reporte de animales por sala	8	80	lun 05/11/2018	lun 05/11/2018
HU_46	Reporte de productos alimenticios	8		mar 06/11/2018	mar 06/11/2018
HU_47	Reporte de productos medicinales	8		mié 07/11/2018	mié 07/11/2018
HU_48	Reporte de las salas dada una fecha	24		jue 08/11/2018	lun 12/11/2018
HU_49	Reporte de ventas por fecha	16		mar 13/11/2018	mié 14/11/2018
HU_50	Reporte de animales en venta	16		jue 15/11/2018	vie 16/11/2018
SPRINT 11					
HT_07	Desarrollo del manual de usuario	40	40	lun 19/11/2018	vie 23/11/2018
TOTAL			840		

Realizado por: Karen Pardo. 2019

Se obtuvo 11 Sprints con un total de 840 horas de trabajo, logrando un proyecto de gran éxito.

2.4.2 Fase de desarrollo

2.4.2.1 Sprint 1

HT_01 Estudio de las herramientas

Para desarrollar el sistema informático se seleccionó herramientas funcionales con rendimiento óptimo para así lograr un proyecto final exitoso. A continuación, se detalla cada una de las herramientas utilizadas.

- **Netbeans 8.2:** Es un entorno de desarrollo integrado (IDE) que trabaja a nivel de proyecto y en el que se pueden realizar tareas que son asociadas a la programación; tales como: editar el código, compilarlo, ejecutarlo y a su vez depurarlo. También ayuda a la simplificación de las tareas tediosas en proyectos grandes.
- **Java EE:** Representa un estándar para la implementación y despliegue de aplicaciones empresariales de cualquier tipo. Esta plataforma de programación fue diseñada a través de un proceso abierto y se entablaron conversaciones con un amplio abanico de empresas, lo que permitió que se cubriesen la gran mayoría de los requisitos de las aplicaciones

empresariales. Las características principales son: modelo multicapa, servidor de aplicaciones, gestión de componentes mediante contenedores, soporte para los clientes, soporte para lógica de negocio y soporte estándar. (Riola y Carlos, 2008, pp. 15-16-17-18-19)

- **Java Server Faces (JSF):** Es un framework web J2EE de código fuente abierto, server-side, basado en componentes de interfaz de usuario, que facilita y agiliza el desarrollo de aplicaciones web que soportan diferentes dispositivos como clientes, por ejemplo: teléfonos celulares, agendas, etc. Los programadores piensan en términos de componentes de interfaz de usuario, eventos, y sus interacciones. JSF oculta al programador los detalles del requerimiento HTTP, la respuesta HTTP y de markups HTML, logrando un estilo de programación similar a la programación de aplicaciones de escritorio, del estilo Java Swing, Delphi o Visual Basic. (Díaz et al., 2007)
- **PostgreSQL 9.4:** Es un sistema gestor de base de datos relacional, que tiene características orientada a objetos y soporta diferentes tipos de datos. Integra una estructura de diversas funciones; tales como: manejo de fechas, orientación de operaciones con redes, etc. Además, permite que se declaren funciones propias y la gestión de los distintos usuarios.
- **Primefaces:** Es una librería de componentes visuales de código abierto para el conjunto Java Server Faces 2.0. El objetivo principal es, ofrecer un conjunto de componentes para facilitar la creación y diseño de aplicaciones web. Además, cuentan con soporte nativo de Ajax y un módulo adicional Touch Faces, que sirve para el desarrollo de aplicaciones web, así como, para dispositivos móviles con navegadores basados en WebKit. (Pech May et al., 2011)
- **Bootstrap:** Es un framework muy utilizado, creado para desarrollar aplicaciones web en el que se incluyen HTML (HyperText Markup Language), CSS (Cascading Style Sheets) y JS (JavaScript). Este marco de trabajo permite que el desarrollo web denominado front – end (del lado del cliente y del lado servidor), se realice de manera más rápida y fácil. Se adapta a todas las formas de los dispositivos, incluyendo computadoras de escritorio, laptops, móviles, tablets, etc. (Veloz, 2016, pp. 31-32)

Tabla 7-2: Historia Técnica 01

HISTORIA TÉCNICA		
Id: HT_01	Nombre: Estudio de las herramientas	
Descripción: Como desarrolladora necesito realizar un estudio profundo sobre las herramientas que se utilizaran en el desarrollo del proyecto.		
Usuario: Desarrolladora	Sprint: 1	
Fecha Inicio: 02/07/2018	Fecha fin: 04/07/2018	
TAREAS DE INGENIERÍA		
ID	NOMBRE	ESFUERZO
TI_01	Selección de herramientas a utilizar	24
Puntos estimados: 24		TOTAL 24

Realizado por: Karen Pardo. 2019

Tabla 8-2: Tarea de Ingeniería 01 de la HT_01

TAREA DE INGENIERÍA		
Id: TI_01	Tipo de tarea: Análisis	Sprint: 1
Nombre de la historia técnica: HT_01 Estudio de las herramientas		
Nombre tarea: Selección de herramientas a utilizar		
Fecha Inicio: 02/07/2018	Fecha fin: 04/07/2018	
Descripción: Como desarrolladora necesito realizar un estudio de las herramientas a utilizarse en el desarrollo del proyecto.		
PRUEBAS DE ACEPTACIÓN		
ID	NOMBRE	
PA_01	Verificar que las herramientas seleccionadas sean las apropiadas para un correcto funcionamiento del sistema.	

Realizado por: Karen Pardo. 2019

Tabla 9-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar que las herramientas seleccionadas sean las apropiadas para un correcto funcionamiento del sistema.
Tarea de ingeniería: Ti_01 Selección de herramientas a utilizar	
Descripción: Verificar que las herramientas a utilizarse sean las necesitadas por el desarrollador.	
Responsable: Karen Pardo	Fecha: 04/07/2018
Condición de ejecución	
<ul style="list-style-type: none"> - Las herramientas deben estar instaladas. 	
Pasos de ejecución	
<ul style="list-style-type: none"> - Verificar la instalación de las herramientas. - Configurar las herramientas. - Verificar el funcionamiento de las herramientas. 	
Resultado esperado: Las herramientas seleccionadas son las correctas para el desarrollo del sistema.	
Evaluación de la prueba: Exitosa	

Realizado por: Karen Pardo. 2019

HT_02 Análisis y gestión de riesgos

La gestión de riesgos es un proceso que ayuda a identificar, determinar y analizar las posibles amenazas a las que se expone un proyecto informático; para lo cual, se planifica una estrategia tomando medidas de precaución, para que no afecte el desarrollo del mismo. De este análisis se identificó ocho posibles riesgos.

Tabla 10-2: Identificación de riesgos

Id	Descripción	Tipo	Consecuencia
R1	Análisis incorrecto previo a las necesidades y requerimientos del proyecto	Del proyecto	Retraso en la ejecución del proyecto debido a cambios en la planificación.
R2	Mal diseño de la base de datos	Del proyecto	Retraso en el desarrollo del proyecto.
R3	Tecnología hardware inadecuada para el desarrollo del proyecto.	Técnico	Incompatibilidad del sistema con el equipo hardware.
R4	Falta de información sobre las herramientas a utilizar.	Técnico	Retraso en el desarrollo del proyecto.
R5	Daño en los equipos debido a desastres naturales.	Técnico	Pérdida total o parcial de información o equipos.
R6	Pérdida de equipos de trabajo.	Técnico	Pérdida total o parcial de información o equipos.
R7	No cumplimiento de los entregables por parte del desarrollador.	Del proyecto	Retraso en la entrega del proyecto.
R8	Falta de comunicación entre el desarrollador y el cliente.	Del negocio	Retraso en los entregables del proyecto.

Realizado por: Karen Pardo. 2019

Análisis de riesgos

Tabla 11-2: Probabilidad del riesgo

Rango de Probabilidades	Descripción	Valor
1% - 33%	Baja	1
34% - 67%	Media	2
68% - 100%	Alta	3




Realizado por: Karen Pardo. 2019

Tabla 12-2: Impacto del riesgo

Impacto	Retraso	Impacto Técnico	Valor
Bajo	1 semana	Ligero efecto en el desarrollo del proyecto	1
Moderado	2 semanas	Moderado efecto en el desarrollo del proyecto	2
Alto	1 mes	Severo efecto en el desarrollo del proyecto	3
Crítico	Más de un mes	El proyecto no puede culminarse	4

Realizado por: Karen Pardo. 2019

Tabla 13-2: Exposición del riesgo

Exposición del Riesgo	Valor	Color
Baja	1 o 2	1 
Media	3 o 4	2 
Alta	Mayor a 6	3 

Realizado por: Karen Pardo. 2019

Tabla 14-2: Impacto probabilidad

Impacto \ Probabilidad	Bajo=1	Moderado=2	Alto=3	Crítico=4
ALTA=3	3	6	9	12
MEDIA=2	2	4	6	8
BAJA=1	1	2	3	4

Realizado por: Karen Pardo. 2019

Tabla 15-2: Priorización de riesgos

Id	Probabilidad			Impacto		Exposición		Prioridad
	Nro.	Porcentaje	Valor	Descripción	Valor	Impacto	Valor	
R1	70%	3	Alta	3	Alto	9	Alta	1
R2	50%	2	Media	3	Alto	6	Alta	2
R3	60%	2	Media	2	Medio	4	Media	5
R4	65%	2	Media	3	Alto	6	Alta	6
R5	20%	1	Baja	3	Alto	3	Alta	7
R6	40%	2	Media	3	Alto	6	Alta	8
R7	10%	1	Baja	2	Medio	2	Baja	4
R8	20%	1	Baja	2	Medio	2	Baja	3

Realizado por: Karen Pardo. 2019

Tabla 16-2: Priorización de riesgos 1

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R1		FECHA: 05/07/2018	
Probabilidad: Alta Valor: 3	Impacto: Alto Valor: 3	Exposición: Alta Valor: 9	Prioridad: 1
DESCRIPCIÓN: Análisis incorrecto previo a las necesidades y requerimientos del proyecto			
REFINAMIENTO			
<u>Causas</u>			
<ul style="list-style-type: none"> - Falta de comunicación - Mala documentación y avance del proyecto 			
<u>Consecuencia</u>			
<ul style="list-style-type: none"> - Retraso en el desarrollo del proyecto - El sistema no cumplirá con todas las funcionalidades requeridas por el cliente 			
REDUCCIÓN			
<ul style="list-style-type: none"> - Realizar una reunión agradable con el cliente para que esté a gusto y poder obtener requerimientos verdaderamente necesarios para implementarlos 			
SUPERVISIÓN			
<ul style="list-style-type: none"> - Mantenerse al tanto de la situación económica y funcional del proyecto 			
GESTIÓN			
<ul style="list-style-type: none"> - Tratar de llegar a un acuerdo con el cliente 			

Realizado por: Karen Pardo. 2019

Tabla 17-2: Priorización de riesgos 2

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R2		FECHA: 05/07/2018	
Probabilidad: Medio Valor: 2	Impacto: Alto Valor: 3	Exposición: Alta Valor: 6	Prioridad: 2
DESCRIPCIÓN: Mal diseño de la base de datos			
REFINAMIENTO			
<u>Causas</u>			
<ul style="list-style-type: none"> - Mal análisis de los requerimientos entregados por el cliente 			

<ul style="list-style-type: none"> - Indecisión por parte del cliente <p><u>Consecuencia</u></p> <ul style="list-style-type: none"> - Retraso en el desarrollo del proyecto
<p>REDUCCIÓN</p> <ul style="list-style-type: none"> - Análisis de los requerimientos correctamente - Ambiente amigable con el cliente
<p>SUPERVISIÓN</p> <ul style="list-style-type: none"> - Analizar las tablas de la base de datos precavidamente
<p>GESTIÓN</p> <ul style="list-style-type: none"> - Consultar con un experto en diseño de base de datos

Realizado por: Karen Pardo. 2019

Tabla 18-2: Priorización de riesgos 3

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R8		FECHA: 05/07/2018	
Probabilidad: Baja Valor: 1	Impacto: Medio Valor: 2	Exposición: Baja Valor: 2	Prioridad: 3
DESCRIPCIÓN: Falta de comunicación entre el desarrollador y el cliente.			
<p>REFINAMIENTO</p> <p><u>Causas</u></p> <ul style="list-style-type: none"> - Malos entendidos entre el desarrollador y el cliente <p><u>Consecuencia</u></p> <ul style="list-style-type: none"> - Mala planificación del proyecto - Retraso en los entregables del proyecto - Retraso en la fecha de finalización del proyecto 			
<p>REDUCCIÓN</p> <ul style="list-style-type: none"> - Tener paciencia a las exigencias del cliente - Mantener responsabilidad tanto cliente como desarrollador 			
<p>SUPERVISIÓN</p> <ul style="list-style-type: none"> - Informar al cliente sobre los avances del desarrollo del proyecto 			
<p>GESTIÓN</p> <ul style="list-style-type: none"> - Mantener reuniones productivas con el cliente 			

Realizado por: Karen Pardo. 2019

Tabla 19-2: Priorización de riesgos 4

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R7		FECHA: 05/07/2018	
Probabilidad: Baja Valor: 1	Impacto: Medio Valor: 2	Exposición: Baja Valor: 2	Prioridad: 4
DESCRIPCIÓN: No cumplimiento de los entregables por parte del desarrollador.			
REFINAMIENTO			
<u>Causas</u>			
<ul style="list-style-type: none"> - Subestimación del tiempo de desarrollo del proyecto - Desconocimiento de las herramientas para el desarrollo 			
<u>Consecuencia</u>			
<ul style="list-style-type: none"> - Retraso en la fecha de finalización del proyecto 			
REDUCCIÓN			
<ul style="list-style-type: none"> - Correcto análisis en la planificación del proyecto 			
SUPERVISIÓN			
<ul style="list-style-type: none"> - Informar al cliente sobre los avances del desarrollo del proyecto 			
GESTIÓN			
<ul style="list-style-type: none"> - Reuniones continuas con el cliente y presentación de los avances 			

Realizado por: Karen Pardo. 2019

Tabla 20-2: Priorización de riesgos 5

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R3		FECHA: 06/07/2018	
Probabilidad: Media Valor: 2	Impacto: Medio Valor: 2	Exposición: Media Valor: 4	Prioridad: 5
DESCRIPCIÓN: Tecnología hardware inadecuada para el desarrollo de proyecto.			
REFINAMIENTO			
<u>Causas</u>			
<ul style="list-style-type: none"> - Hardware incompatible con las aplicaciones de desarrollo 			
<u>Consecuencia</u>			
<ul style="list-style-type: none"> - Errores en la instalación de software para el desarrollo del sistema - Errores de conexión 			

REDUCCIÓN
- Realizar pruebas de funcionamiento de las aplicaciones a utilizar para el desarrollo del sistema
SUPERVISIÓN
- Verificar la capacidad de almacenamiento de los equipos de desarrollo
GESTIÓN
- Adquirir equipos de buenas características

Realizado por: Karen Pardo. 2019

Tabla 21-2: Priorización de riesgos 6

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R4		FECHA: 06/07/2018	
Probabilidad: Media Valor: 2	Impacto: Alto Valor: 3	Exposición: Alta Valor: 6	Prioridad: 6
DESCRIPCIÓN: Falta de información sobre las herramientas a utilizar.			
REFINAMIENTO			
<u>Causas</u>			
- Desconocimiento de las herramientas a ser utilizadas para el desarrollo del proyecto			
<u>Consecuencia</u>			
- Errores de conexión			
- Instalación de herramientas inapropiadas			
REDUCCIÓN			
- Profundizar en la investigación sobre las herramientas a utilizar			
SUPERVISIÓN			
- Verificar que las aplicaciones a utilizar sean adecuadas para el desarrollo del proyecto			
GESTIÓN			
- Utilizar herramientas actualizadas y con licencia			

Realizado por: Karen Pardo. 2019

Tabla 22-2: Priorización de riesgos 7

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R5		FECHA: 06/07/2018	
Probabilidad: Baja	Impacto: Alto	Exposición: Alta	Prioridad: 7

Valor: 1	Valor: 3	Valor: 3	
DESCRIPCIÓN: Daño en los equipos debido a desastres naturales.			
REFINAMIENTO			
<u>Causas</u>			
<ul style="list-style-type: none"> - Temblores - Incendios - Erupciones de volcanes 			
<u>Consecuencia</u>			
<ul style="list-style-type: none"> - Pérdida parcial o total de información - Pérdidas económicas para el negocio 			
REDUCCIÓN			
<ul style="list-style-type: none"> - Tener una infraestructura resistente - Tomar medidas de precaución 			
SUPERVISIÓN			
<ul style="list-style-type: none"> - Verificar que los extintores estén en buen estado - Verificar que existan respaldos del sistema y base de datos 			
GESTIÓN			
<ul style="list-style-type: none"> - Guardar respaldos del sistema y de la base de datos 			

Realizado por: Karen Pardo. 2019

Tabla 23-2: Priorización de riesgos 8

HOJA DE GESTIÓN DE RIESGOS			
ID. DEL RIESGO: R6		FECHA: 06/07/2018	
Probabilidad: Media Valor: 2	Impacto: Alto Valor: 3	Exposición: Alta Valor: 6	Prioridad: 8
DESCRIPCIÓN: Pérdida de equipos de trabajo.			
REFINAMIENTO			
<u>Causas</u>			
<ul style="list-style-type: none"> - Asalto al personal de desarrollo 			
<u>Consecuencia</u>			
<ul style="list-style-type: none"> - Pérdida parcial o total de información - Reimplantación del sistema 			
REDUCCIÓN			
<ul style="list-style-type: none"> - Tomar precauciones para la seguridad del personal y de los equipos de desarrollo 			
SUPERVISIÓN			

- Verificar la seguridad del personal y de los equipos de desarrollo
GESTIÓN
- Guardar respaldos del sistema y de la base de datos

Realizado por: Karen Pardo. 2019

Definidos todos los riesgos a los cuales está expuesto el proyecto, se realizó su historia técnica, tarea de ingeniería y la prueba de aceptación, las cuales están detalladas a continuación.

Tabla 24-2: Historia Técnica 02

HISTORIA TÉCNICA		
Id: HT_02	Nombre: Análisis y gestión de riesgos.	
Descripción: Es preciso identificar las amenazas, así como también, la probabilidad de que ocurran y su impacto en el desarrollo del proyecto; esto se realiza siempre para determinar los riesgos, poder definirlos y saber a cuáles debemos enfrentarnos, y así tomar las debidas precauciones.		
Usuario: Desarrolladora	Sprint: 1	
Fecha Inicio: 05/07/2018	Fecha fin: 06/07/2018	
TAREAS DE INGENIERÍA		
ID	NOMBRE	ESFUERZO
TI_01	Identificación y probabilidad de los riesgos	16
Puntos estimados: 16	TOTAL	16

Realizado por: Karen Pardo. 2019

Tabla 25-2: Tarea de Ingeniería 01 de la HT_02

TAREA DE INGENIERÍA		
Id: TI_01	Tipo de tarea: Análisis	Sprint: 1
Nombre de la historia técnica: HT_02 Análisis y gestión de riesgos		
Nombre tarea: Identificación y probabilidad de los riesgos		
Fecha Inicio: 05/07/2018	Fecha fin: 06/07/2018	
Descripción: Como desarrolladora necesito analizar los posibles riesgos que se pueden presentar en el desarrollo del proyecto informático.		
PRUEBAS DE ACEPTACIÓN		
ID	NOMBRE	
PA_01	Verificar los posibles riesgos y amenazas que se pueden presentar en el desarrollo del proyecto informático.	

Realizado por: Karen Pardo. 2019

Tabla 26-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar los posibles riesgos y amenazas que se pueden presentar en el desarrollo del proyecto informático.
Tarea de ingeniería: TI_01 Identificación y probabilidad de los riesgos	

Descripción: Verificar cada uno de los riesgos al que se encuentra expuesto el proyecto informático, con el fin de poder realizar la gestión del mismo.	
Responsable: Karen Pardo	Fecha: 06/07/2018
Condición de ejecución	
<ul style="list-style-type: none"> - Identificar roles y usuarios - Exista la planificación del proyecto. 	
Pasos de ejecución	
<ul style="list-style-type: none"> - Verificar los riesgos que pueden existir en cada uno de los usuarios. - Revisar la planificación del proyecto. 	
Resultado esperado: Los posibles riesgos se han identificado y se han establecido las debidas medidas correctivas.	
Evaluación de la prueba: Exitosa	

Realizado por: Karen Pardo. 2019

HT_03 Definición del estándar de codificación

El estándar de codificación es parte de las llamadas buenas o mejores prácticas, este es un conjunto no formal de reglas, que han ido surgiendo en las distintas comunidades de desarrolladores, las cuales pueden incrementar la calidad del código de programación. (Merkury, 2017)

El estándar de codificación se utiliza para ordenar el código del sistema, de tal forma que los desarrolladores puedan entenderlo en menos tiempo. Por lo tanto, para el desarrollo del sistema de venta y crianza de cerdos se utiliza el estándar de codificación UpperCamelCase y el lowerCamelCase.

Características de los estándares:

- UpperCamelCase. - La primera letra de cada palabra debe siempre estar en mayúscula, por ejemplo: BaseDatos.
- lowerCamelCase. - Se aplica lo mismo que la anterior, pero con la excepción de que la primera letra debe estar en minúscula, por ejemplo: creacionBaseDatos.
- No deben ir puntos, guion medio o guion bajo entre las palabras, por ejemplo: creacion_BaseDatos.

A continuación, se realiza la historia técnica, tarea de ingeniería y la prueba de aceptación de la definición del estándar de codificación.

Tabla 27-2: Historia Técnica 03

HISTORIA TÉCNICA		
Id: HT_03	Nombre: Definición del estándar de codificación	
Descripción: Como desarrolladora necesito establecer un estándar de codificación para el desarrollo del sistema		
Usuario: Desarrolladora	Sprint: 1	
Fecha Inicio: 09/07/2018	Fecha fin: 11/07/2018	
TAREAS DE INGENIERÍA		
ID	NOMBRE	ESFUERZO
TI_01	Investigación del estándar de codificación a utilizar	24
Puntos estimados: 24		TOTAL 24

Realizado por: Karen Pardo. 2019

Tabla 28-2: Tarea de Ingeniería 01 de la HT_03

TAREA DE INGENIERÍA		
Id: TI_01	Tipo de tarea: Desarrollo	Sprint: 1
Nombre de la historia técnica: HT_03 Definición del estándar de codificación		
Nombre tarea: Investigación del estándar de codificación a utilizar		
Fecha Inicio: 09/07/2018	Fecha fin: 11/07/2018	
Descripción: Como desarrolladora necesito investigar sobre el mejor estándar para aplicar en el sistema.		
PRUEBAS DE ACEPTACIÓN		
ID	NOMBRE	
PA_01	Verificar si el estándar de codificación se puede utilizar en el sistema.	

Realizado por: Karen Pardo. 2019

Tabla 29-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar si el estándar de codificación se puede utilizar en el sistema.
Tarea de ingeniería: TI_01 Definición del estándar de codificación	
Descripción: Verificar si el estándar de codificación se puede aplicar en la creación de las tablas en la base de datos.	
Responsable: Karen Pardo	Fecha: 11/07/2018
Condición de ejecución	
- Conocer perfectamente sobre el estándar de codificación que se va aplicar.	
Pasos de ejecución	
- Verificar que el estándar sea aplicable en el sistema informático.	
Resultado esperado: El estándar de codificación si es el apropiado para el desarrollo del sistema informático.	
Evaluación de la prueba: Exitosa	

Realizado por: Karen Pardo. 2019

HT_04 Diseño de la Base de Datos

En el diseño de la base de datos se determinan las entidades, relaciones y atributos que forman parte del sistema informático. Una estructura correcta nos da acceso a la información actualizada del mismo.

Es importante analizar los requerimientos que da el cliente, para luego implementar correctamente la base de datos y no tener ningún inconveniente en el desarrollo del proyecto.

Se implementa el diseño lógico vinculado en el **ANEXO B** donde se definen las entidades del negocio, el diseño físico y el detalle de las tablas. Además, se realiza el diccionario de datos ubicado en el **ANEXO C**, el cual define los tipos de datos de los campos de cada una de las tablas, la relación que existe entre cada una, las claves primarias y la longitud de cada campo. En total se definieron 16 tablas.

A continuación, se realiza la historia técnica, las tareas de ingeniería y la prueba de aceptación del diseño de la base de datos.

Tabla 30-2: Historia Técnica 04

HISTORIA TÉCNICA		
Id: HT_04	Nombre: Diseño de la Base de Datos.	
Descripción: Como desarrolladora necesito elaborar el Diseño de la Base de Datos con la que se trabajará en el desarrollo del sistema informático.		
Usuario: Desarrolladora	Sprint: 1 y 2	
Fecha Inicio: 12/07/2018	Fecha fin: 18/07/2018	
TAREAS DE INGENIERÍA		
ID	NOMBRE	ESFUERZO
TI_01	Realizar diseño lógico de la base de datos	24
TI_02	Realizar el diccionario de datos	16
Puntos estimados: 40		TOTAL 40

Realizado por: Karen Pardo. 2019

Tabla 31-2: Tarea de Ingeniería 01 de la HT_04

TAREA DE INGENIERÍA		
Id: TI_01	Tipo de tarea: Diseño	Sprint: 1 y 2
Nombre de la historia técnica: HT_04 Diseño de la Base de Datos.		
Nombre tarea: Realizar diseño lógico de la base de datos		
Fecha Inicio: 12/07/2018	Fecha fin: 14/07/2018	
Descripción: Como desarrolladora necesito saber todos los requerimientos del cliente para realizar el diseño lógico de la base de datos.		

PRUEBAS DE ACEPTACIÓN	
ID	NOMBRE
PA_01	Verificar que la base de datos este bien estructurada.

Realizado por: Karen Pardo. 2019

Tabla 32-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar que la base de datos este bien estructurada.
Tarea de ingeniería: TI_01 Realizar diseño lógico de la base de datos	
Descripción: Verificar que no exista redundancia de datos dentro de la base.	
Responsable: Karen Pardo	Fecha: 14/07/2018
Condición de ejecución	
- Los requerimientos deben estar planteados	
Pasos de ejecución	
- Revisar que la base de datos creada cumpla con todos los requerimientos.	
Resultado esperado: La base de datos si cumple con lo requerido	
Evaluación de la prueba: Exitosa	

Realizado por: Karen Pardo. 2019

Tabla 33-2: Tarea de Ingeniería 02 de la HT_04

TAREA DE INGENIERÍA		
Id: TI_02	Tipo de tarea: Diseño	Sprint: 1 y 2
Nombre de la historia técnica: HT_04 Diseño de la Base de Datos.		
Nombre tarea: Realizar el diccionario de datos		
Fecha Inicio: 17/07/2018	Fecha fin: 18/07/2018	
Descripción: Como desarrolladora necesito definir perfectamente el diccionario de datos para facilitar el desarrollo del sistema.		
PRUEBAS DE ACEPTACIÓN		
ID	NOMBRE	
PA_01	Verificar que los tipos de datos sean los apropiados.	

Realizado por: Karen Pardo. 2019

Tabla 34-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar que los tipos de datos sean los apropiados
Tarea de ingeniería: TI_02 Realizar el diccionario de datos.	
Descripción: Verificar que los tipos de datos de la base sean los apropiados.	
Responsable: Karen Pardo	Fecha: 18/07/2018
Condición de ejecución	
- Deben estar todos los requerimientos planteados.	
Pasos de ejecución	
- Revisar que los tipos de datos de cada una de las tablas.	

Resultado esperado: El diccionario de datos si cumple con todo lo establecido.
Evaluación de la prueba: Exitosa

Realizado por: Karen Pardo. 2019

2.4.2.2 Sprint 2

HT_05 Diseño de la arquitectura del sistema

Para el desarrollo e implementación del sistema informático se utiliza la arquitectura Modelo – Vista – Controlador (MVC), la cual se clasifica en tres carpetas diferentes: En la carpeta del Modelo se realiza todo lo referente al acceso de la información, para poder conectarse con la base de datos; en la carpeta Controlador se almacenan los archivos que permiten la comunicación entre la vista y el modelo; y en la carpeta Vista se guarda todo lo referente a la interfaz del sistema con la que el usuario interactúa.

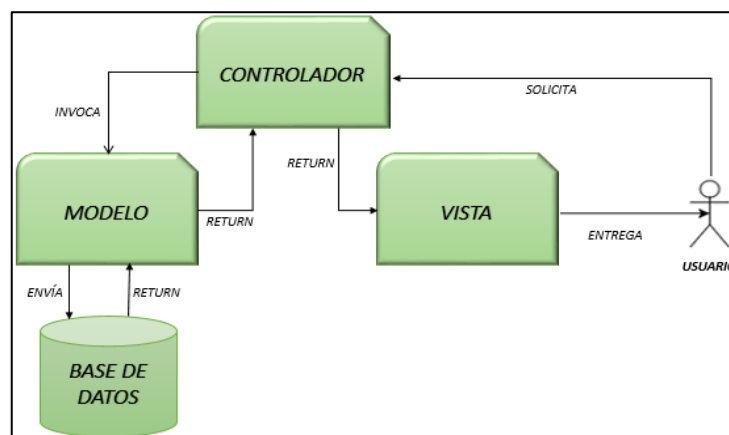


Figura 1-2: Arquitectura del sistema

Realizado por: Karen Pardo. 2019

En la **Figura 1-2** se presenta la arquitectura del sistema en donde el usuario solicita al controlador información, el controlador invoca al modelo por dicha información, el modelo envía la petición a la base de datos, ésta retorna la información solicitada al modelo, el modelo entrega la información al controlador, y éste entrega lo solicitado al usuario por medio de la vista.

A continuación, se realiza la historia técnica, tarea de ingeniería y la prueba de aceptación del diseño de la arquitectura del sistema.

Tabla 35-2: Historia Técnica 05

HISTORIA TÉCNICA		
Id: HT_05	Nombre: Diseño de la arquitectura del sistema.	
Descripción: Como desarrolladora necesito detallar la arquitectura con la que se implementará el sistema informático.		
Usuario: Desarrolladora	Sprint: 2	
Fecha Inicio: 19/07/2018	Fecha fin: 23/07/2018	
TAREAS DE INGENIERÍA		
ID	NOMBRE	ESFUERZO
TI_01	Realizar el diseño de la arquitectura del sistema	24
Puntos estimados: 24		TOTAL 24

Realizado por: Karen Pardo. 2019

Tabla 36-2: Tarea de Ingeniería 01 de la HT_05

TAREA DE INGENIERÍA		
Id: TI_01	Tipo de tarea: Desarrollo	Sprint: 2
Nombre de la historia técnica: HT_05 Diseño de la arquitectura del sistema.		
Nombre tarea: Realizar el diseño de la arquitectura del sistema		
Fecha Inicio: 19/07/2018	Fecha fin: 23/07/2018	
Descripción: Como desarrolladora necesito realizar el diseño de la arquitectura del sistema.		
PRUEBAS DE ACEPTACIÓN		
ID	NOMBRE	
PA_01	Verificar que los componentes de la arquitectura del sistema sean los adecuados.	

Realizado por: Karen Pardo. 2019

Tabla 37-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar que los componentes de la arquitectura del sistema sean los adecuados.
Tarea de ingeniería: TI_01 Realizar el diseño de la arquitectura del sistema	
Descripción: Verificar que todos los componentes del sistema estén bien implementados.	
Responsable: Karen Pardo	Fecha: 23/07/2018
Condición de ejecución	
- La arquitectura del sistema debe estar diseñada.	
Pasos de ejecución	
- Revisar que estén bien implementados todos los componentes de la arquitectura del sistema.	
Resultado esperado: La arquitectura está bien diseñada.	
Evaluación de la prueba: Exitosa	

Realizado por: Karen Pardo. 2019

HT_06 Diseño de las interfaces de usuario

En esta historia técnica se presenta de forma precisa el diseño de las interfaces de usuario para el desarrollo del sistema, definiendo: la estructura, colores, fuentes, formas y tamaño de los objetos, etc. Esto, permite que el sistema sea funcional y de buen agrado para los usuarios.

En las **Figuras 2 -2, 3 -2 y 4 -2**, se puede observar cómo está estructurada la interfaz de usuario de la página principal del sistema, la cual está dividida en tres secciones que son: cabecera, cuerpo y pie de página.



Figura 2-2: Cabecera de la página principal del sistema.

Realizado por: Karen Pardo. 2019

En la **Figura 2-2** se muestra la **cabecera**, la cual presenta el nombre de la granja y un menú que direcciona a otras interfaces que dan ingreso al sistema, información de los antecedentes, contactos y raza de cerdos de la granja.

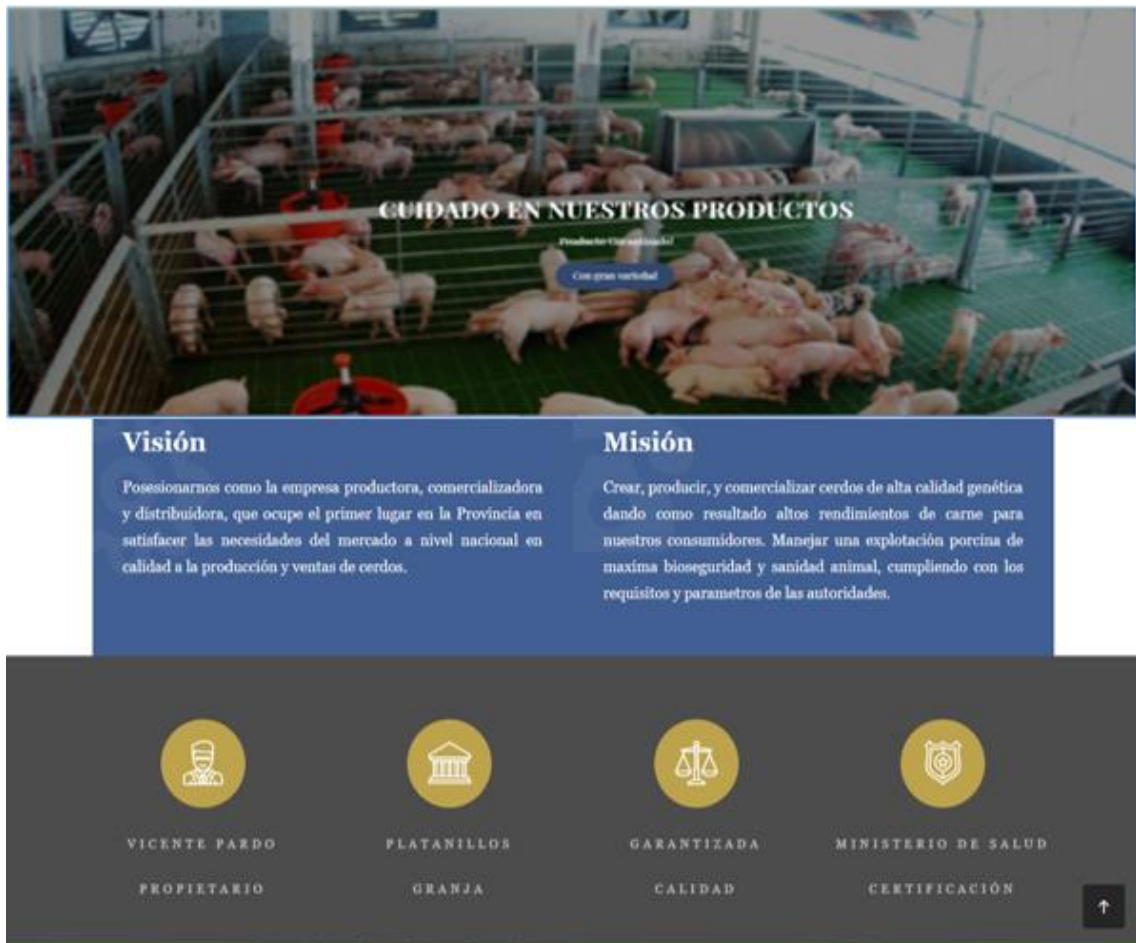


Figura 3-2: Cuerpo de la página principal del sistema.

Realizado por: Karen Pardo. 2019

En la **Figura 3-2** se muestra el **cuerpo**, el cual presenta imágenes de las salas de crianza de cerdos, la Visión y Misión de la granja; así como también, el nombre del propietario, nombre de la granja, la calidad del producto y la certificación del Ministerio de Salud.

Navegación	Información	Horario de atención
Inicio	EL Oro - Piñas - Platanillos	Lunes - Viernes: 8:00 - 18:00
Antecedentes	0979424405	Sábado - Domingo 8:00 - 17:00
Contacto	gplatanillos@hotmail.com	
Razas		

© 2019. Todos los derechos reservados.

Figura 4-2: Pie de la página principal del sistema.

Realizado por: Karen Pardo. 2019

En la **Figura 4-2** se muestra el **pie de página**, el cual presenta un menú de navegación similar a la cabecera, los derechos de autor, ubicación, número de teléfono y correo electrónico del propietario de la granja; así como también, los horarios de atención.

En las **Figuras 5 -2, 6 -2 y 7 -2**, se puede observar cómo está estructurada la interfaz de usuario de la página principal de ingreso al sistema, la cual está dividida en tres secciones que son: cabecera, cuerpo y pie de página.



Figura 5-2: Cabecera de la página principal de ingreso al sistema.

Realizado por: Karen Pardo. 2019

En la **Figura 5 -2** se muestra la **cabecera**, la cual presenta el nombre de la granja y un botón para desplegar el menú.

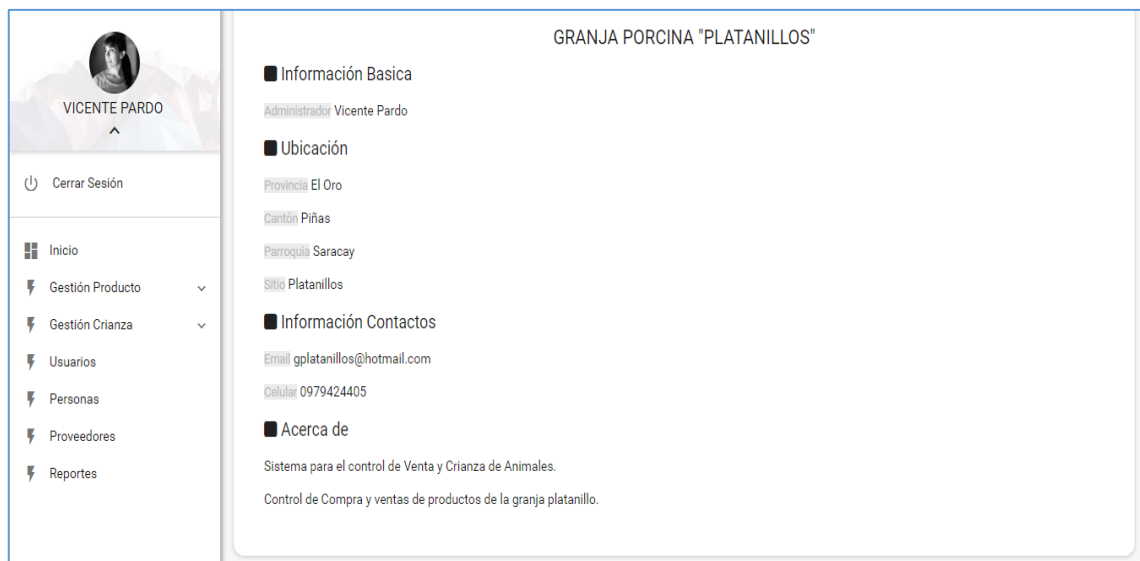


Figura 6-2: Cuerpo de la página principal de ingreso al sistema.

Realizado por: Karen Pardo. 2019

En la **Figura 6 -2** se muestra el **cuerpo**, el cual presenta el nombre del usuario, un menú de los procesos del sistema, el tipo de usuario, el nombre, ubicación y contactos de la granja.



Figura 7-2: Pie de la página principal de ingreso al sistema.

Realizado por: Karen Pardo. 2019

En la **Figura 7 -2** se muestra el **pie de página**, el cual presenta el logo de la granja, así como también, los derechos de autor.

A continuación, se realiza la historia técnica, tarea de ingeniería y la prueba de aceptación del diseño de las interfaces de usuario.

Tabla 38-2: Historia Técnica 06

HISTORIA TÉCNICA		
Id: HT_06	Nombre: Diseño de las interfaces de usuario	
Descripción: Como desarrolladora necesito definir el diseño de las interfaces de usuario conjuntamente con el cliente para facilitar el desarrollo del sistema.		
Usuario: Desarrolladora	Sprint: 2 y 3	
Fecha Inicio: 24/07/2018	Fecha fin: 30/07/2018	
TAREAS DE INGENIERÍA		
ID	NOMBRE	ESFUERZO
TI_01	Diseño de la interfaz de usuario del sistema.	40
Puntos estimados: 40		TOTAL 40

Realizado por: Karen Pardo. 2019

Tabla 39-2: Tarea de Ingeniería 01 de la HT_06

TAREA DE INGENIERÍA		
Id: TI_01	Tipo de tarea: Desarrollo	Sprint: 2 y 3
Nombre de la historia técnica: HT_06 Diseño de la arquitectura del sistema.		
Nombre tarea: Diseño de la interfaz de usuario del sistema.		
Fecha Inicio: 24/07/2018	Fecha fin: 30/07/2018	
Descripción: Como desarrolladora necesito realizar un diseño de interfaz de usuario para facilitar el desarrollo del sistema.		
PRUEBAS DE ACEPTACIÓN		
ID	NOMBRE	
PA_01	Verificar que el diseño de la interfaz de usuario cumpla con los requisitos del cliente.	

Realizado por: Karen Pardo. 2019

Tabla 40-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar que el diseño de la interfaz de usuario cumpla con los requisitos del cliente.
Tarea de ingeniería: TI_01 Diseño de la interfaz de usuario del sistema.	
Descripción: Verificar que el diseño de la interfaz de usuario cumpla con los requisitos del cliente.	
Responsable: Karen Pardo	Fecha: 30/07/2018
Condición de ejecución	
- Debe estar diseñada la interfaz de usuario.	

Pasos de ejecución
- Verificar que la interfaz cumpla con todos los estándares ya establecidos.
Resultado esperado: El diseño de la interfaz si cumple con todos los estándares ya establecidos.
Evaluación de la prueba: Exitosa

Realizado por: Karen Pardo. 2019

HT_07 Desarrollo del manual de usuario

El manual de usuario se desarrolla para automatizar el proceso de control de venta y crianza de cerdos de la granja porcina; además, sirve de ayuda para los usuarios que no saben el manejo del sistema informático. Éste manual simplifica el tiempo de realización de los procesos en la granja y está adjunto al trabajo de titulación.

A continuación, se realiza la historia técnica, tarea de ingeniería y la prueba de aceptación del manual de usuario.

Tabla 41-2: Historia Técnica 07

HISTORIA TÉCNICA		
Id: HT_07	Nombre: Desarrollo del manual de usuario.	
Descripción: Como desarrolladora necesito realizar el manual de usuario del sistema informático, este ayudará a que se conozcan todas las funcionalidades del sistema de venta y crianza de cerdos, se lo adjuntara en un CD junto al trabajo de titulación.		
Usuario: Desarrolladora	Sprint: 11	
Fecha Inicio: 19/11/2018	Fecha fin: 23/11/2018	
TAREAS DE INGENIERÍA		
ID	NOMBRE	ESFUERZO
TI_01	Desarrollar el manual de usuario del sistema informático	
Puntos estimados:		TOTAL

Realizado por: Karen Pardo. 2019

Tabla 42-2: Tarea de Ingeniería 01 de la HT_07

TAREA DE INGENIERÍA		
Id: TI_01	Tipo de tarea: Desarrollo	Sprint: 11
Nombre de la historia técnica: HT_07 Desarrollo del manual de usuario.		
Nombre tarea: Desarrollar el manual de usuario del sistema informático.		
Fecha Inicio: 19/11/2018	Fecha fin: 23/11/2018	
Descripción: Como desarrolladora necesito crear el manual de usuario para conocer perfectamente las funcionalidades del sistema informático.		
PRUEBAS DE ACEPTACIÓN		
ID	NOMBRE	
PA_01	Verificar que el manual de usuario sea completamente descriptivo.	

Realizado por: Karen Pardo. 2019

Tabla 43-2: Prueba de aceptación 01

PRUEBA DE ACEPTACIÓN	
Id: PA_01	Nombre: Verificar que el manual de usuario sea completamente descriptivo.
Tarea de ingeniería: TI_01 Desarrollar el manual de usuario del sistema informático	
Descripción: El manual deberá tener descripción de cada una de las pantallas	
Responsable: Karen Pardo	Fecha: 23/11/2018
Condición de ejecución <ul style="list-style-type: none">- Debe estar terminado el sistema informático.- Debe estar creado el manual de usuario	
Pasos de ejecución <ul style="list-style-type: none">- Ingresar al sistema informático- Seguir cada paso junto con el manual de usuario- Verificar que cada paso se vaya realizando tanto en el sistema como en el manual de usuario.	
Resultado esperado: El manual de usuario está correctamente realizado	
Evaluación de la prueba: Exitosa	

Realizado por: Karen Pardo. 2019

2.4.3 Fase de cierre

Para esta fase se realiza reuniones con el administrador de la granja porcina “Platanillos” para verificar el cumplimiento de todos los requerimientos y funcionalidades del sistema, logrando así dar finalización al proyecto informático.

2.4.3.1 Avance del proyecto

Para dar avances del proyecto se realizan reuniones con el administrador de la granja al finalizar cada sprint, y se entrega lo planificado de esa etapa. Además, para demostrar el esfuerzo de la cantidad de trabajo que se entrega en cada Sprint, se realiza el grafico Burn Down Chart, en el cual se muestra el esfuerzo total y la velocidad del equipo, para determinar el cumplimiento de los objetivos que se plantearon desde el inicio.

Durante el desarrollo del proyecto no se presentan inconvenientes que afecten la realización del mismo; para determinar el tiempo de desarrollo se realiza un análisis, en los cuales intervienen los tiempos estimados y los tiempos reales que se dan en cada uno de los sprints.

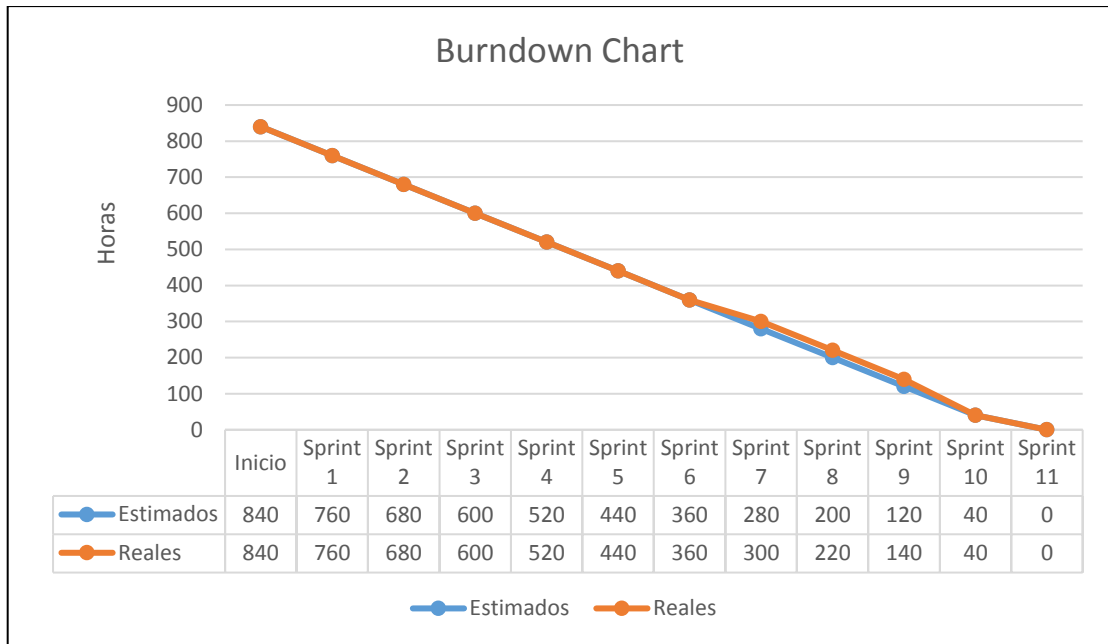


Gráfico 1-2: Burn Down Chart
Realizado por: Karen Pardo. 2019

Un sprint tiene 80 puntos estimados, el cual pertenece a 80 horas laborables de trabajo. En el **Gráfico 1- 2** se puede observar que en el sprint 7 se realiza una sobrestimación de puntos, por lo cual, el proyecto avanza más rápido, ya que en el sprint 7 solo se utiliza 60 puntos de los 80 que fueron estimados. Por el contrario, en el sprint 10 existe una subestimación de 20 puntos, debido a que, en este sprint se realizan los reportes pertinentes; por tal motivo, se aprovechan los 20 puntos que se sobrestimaron en el sprint 7.

Así se concluye con el proyecto apropiadamente, cumpliendo con todos los requerimientos que el cliente solicita y con los días planificados.

CAPÍTULO III

3 RESULTADOS Y DISCUSIÓN

En éste capítulo se puntualiza el análisis de resultados, los cuales se obtuvieron mediante la aplicación de entrevistas dirigidas al administrador y a los empleados de la granja porcina “Platanillos”, por medio de éstas entrevistas se obtuvieron resultados que ayudaron a evaluar el nivel de usabilidad y eficiencia del sistema, en comparación a los tiempos de los procesos de venta y crianza de cerdos de forma tradicional. En el apartado **2.3 Técnicas de investigación**, se explica el modelo de la entrevista y la medición de los tiempos que se realizaron para obtener los resultados.

Actualmente, la granja porcina “Platanillos” cuenta con un administrador y cinco empleados que forman parte de la misma, con la finalidad de alcanzar un resultado del 100%, se aplicaron las entrevistas ya mencionadas anteriormente a todos los usuarios que utilizarán el sistema para el control de venta y crianza de cerdos.

3.1 Resultado de medición de indicadores

3.1.1 Evaluación de la Usabilidad

Para medir la usabilidad del sistema informático se aplicaron entrevistas basadas en la *escala de Likert*; las preguntas planteadas en las entrevistas sirvieron para evaluar los parámetros de: Comprensibilidad, que corresponden a las preguntas 1, 2 y 3; Facilidad de aprendizaje, que corresponden a las preguntas 4 y 5; Operabilidad, que corresponden a las preguntas 6, 7 y 8; y Atractivo, que corresponden a las preguntas 9 y 10.

La siguiente **Tabla 1-3** ayudó a verificar el nivel de usabilidad del sistema en cuanto a la valoración de los niveles y la actitud. Esta tabla se utilizó para todas las preguntas excepto la pregunta número 5, la cual utilizó la **Tabla 2-3**.

Tabla 1-3: Tipo de escala de medición de Likert

Positivo		Negativo		
Siempre	Casi Siempre	Algunas veces	Muy pocas veces	Nunca
4	3	2	1	0

Realizado por: Karen Pardo. 2019

Tabla 2-3: Tipo de escala de medición de Likert a la inversa

Positivo		Negativo		
Nunca	Muy pocas veces	Algunas veces	Casi Siempre	Siempre
4	3	2	1	0

Realizado por: Karen Pardo. 2019

En la **Tabla 3-3** para calcular la escala de usabilidad del sistema, se utilizó la siguiente fórmula:

$$\text{Porcentaje de Usabilidad} = \frac{(\sum_{i=0}^{n=4} TUNi * i) * 100\%}{24}$$

$$\text{Usabilidad al 100\%} = \frac{(6(4)+0(3)+0(2)+0(1)+0(0)) * 100\%}{24} = 100\%$$

$$\text{Usabilidad al 75\%} = \frac{(0(4)+6(3)+0(2)+0(1)+0(0)) * 100\%}{24} = 75\%$$

$$\text{Usabilidad al 50\%} = \frac{(0(4)+0(3)+6(2)+0(1)+0(0)) * 100\%}{24} = 50\%$$

$$\text{Usabilidad al 25\%} = \frac{(0(4)+0(3)+0(2)+6(1)+0(0)) * 100\%}{24} = 25\%$$

$$\text{Usabilidad al 0\%} = \frac{(0(4)+0(3)+0(2)+0(1)+6(0)) * 100\%}{24} = 0\%$$

TUN= Total de Usuarios Nivel

i= Variable de incremento

Tabla 3-3: Cálculos para obtener la escala de Usabilidad

N° de usuarios	Valor del nivel	N° de personas * Valor del nivel	Porcentaje
6	4	24	100%
6	3	18	75%
6	2	12	50%
6	1	6	25%
6	0	0	0%

Realizado por: Karen Pardo. 2019

Los resultados que se obtuvieron se los interpreta mediante la escala presentada en la siguiente tabla.

Tabla 4-3: Escala de Usabilidad

Porcentaje	Resultado
0% a 25%	Deficiente
26% a 50%	Medianamente aceptable
51% a 75%	Aceptable
76% a 100%	Usable

Realizado por: Karen Pardo. 2019

Para calcular el porcentaje de usabilidad de la medición de los indicadores se utilizó la siguiente fórmula:

$$\text{Porcentaje de Usabilidad Indicadores} = \frac{(\sum_{i=0}^{n=4} TUNi * i) * 100\%}{24}$$

TUN= Total de Usuarios Nivel

i= Variable de incremento

3.1.1.1 Medición de los Indicadores utilizando el sistema

Pregunta N° 1: Facilidad del manejo del sistema

Tabla 5-3: Facilidad del manejo del sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	3	2	0	0	0	5
Total usuarios	4	2	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	16	6	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(4(4)+2(3)+0(2)+0(1)+0(0))*100\%}{24} = 91.67\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la facilidad del manejo del sistema, se obtuvo que el 91.67% no tuvo ningún inconveniente al manejar el sistema informático; puesto que los procesos de ingreso de animales a las salas, de ventas de cerdos y control de productos se muestran de manera específica y de fácil manejo por parte del usuario.

Pregunta N° 2: Comprensión de elementos en la interfaz

Tabla 6-3 Comprensión de elementos en la interfaz

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	3	1	1	0	0	5
Total usuarios	4	1	1	0	0	6
<i>Total usuarios x Total valor nivel</i>	16	3	2	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(4(4)+1(3)+1(2)+0(1)+0(0))*100\%}{24} = 87.50\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la comprensión de los elementos en la interfaz del sistema, se obtuvo que el 87.50% no tuvo ningún inconveniente en cuanto a la verificación de los elementos que conforman las interfaces (menú principal, submenús, botones de comando, etc.); dado que éstos se mostraban de manera clara y directa en el proceso que el usuario ejecutaba.

Pregunta N° 3: Información de ubicación del usuario en el sistema

Tabla 7-3: Información de ubicación del usuario en el sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	4	1	0	0	0	5
Total usuarios	5	1	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	20	3	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(5(4)+1(3)+0(2)+0(1)+0(0))*100\%}{24} = 95.83\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la información para entender donde está ubicado el usuario en el sistema, se obtuvo que el 95.83% no tuvo ningún inconveniente de entender su ubicación; puesto que, el sistema a través de la interfaz le especifica al usuario en que proceso está ubicado.

Pregunta N° 4: Facilidad de uso del sistema

Tabla 8-3: Facilidad de uso del sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	3	2	0	0	0	5
Total usuarios	4	2	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	16	6	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(4(4)+2(3)+0(2)+0(1)+0(0))*100\%}{24} = 91.67\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la facilidad de uso del sistema, se obtuvo que el 91.67% no tuvo ninguna dificultad al utilizar el sistema informático; gracias a la

implementación del manual de usuario el cual facilitó el manejo del sistema informático a las personas entrevistadas.

Pregunta N° 5: Complejidad del sistema

Tabla 9-3: Complejidad del sistema

<i>Usuarios</i>	<i>Nunca (4)</i>	<i>Muy pocas veces (3)</i>	<i>Algunas veces (2)</i>	<i>Casi Siempre (1)</i>	<i>Siempre (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	4	1	0	0	0	5
Total usuarios	5	1	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	20	3	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(0(0)+0(1)+0(2)+1(3)+5(4))*100\%}{24} = 95.83\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la complejidad del sistema, se obtuvo que el 95.83% no tuvo ningún tipo de complejidad; dado que el manual de usuario muestra de forma clara y sencilla los pasos necesarios para ejecutar un proceso específico en el sistema informático.

Pregunta N° 6: Mensajes de error mostrados por el sistema al usuario.

Tabla 10-3: Mensajes de error mostrados por el sistema al usuario.

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	5	0	0	0	0	5
Total usuarios	6	0	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	24	0	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(6(4)+0(3)+0(2)+0(1)+0(0))*100\%}{24} = 100\%$$

Después de haber realizado los cálculos pertinentes en cuanto a los mensajes de errores mostrados por el sistema al usuario, se obtuvo que el 100% manifestó que si existe la presentación de errores que el usuario comete cuando completa un formulario; gracias a la correcta implementación de estos mensajes en el código del sistema.

Pregunta N° 7: Integración de las funciones del sistema

Tabla 11-3: Integración de las funciones del sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	4	1	0	0	0	5
Total usuarios	5	1	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	20	3	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(5(4)+1(3)+0(2)+0(1)+0(0))*100\%}{24} = 95.83\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la integración de las funciones del sistema, se obtuvo que el 95.83% manifestó que las funciones están integradas correctamente; puesto que la interfaz mostraba la información del proceso (ventas, ingreso de animales, control de productos, stock de animales, etc.), que el usuario solicitaba al sistema informático.

Pregunta N° 8: Sencillez al buscar y filtrar información dentro del sistema

Tabla 12-3: Sencillez al buscar y filtrar información dentro del sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	4	1	0	0	0	5
Total usuarios	5	1	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	20	3	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(5(4)+1(3)+0(2)+0(1)+0(0))*100\%}{24} = 95.83\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la sencillez al buscar y filtrar información dentro del sistema, se obtuvo que el 95.83% manifestó que si existía sencillez al buscar y filtrar información; dado que el usuario pudo encontrar de manera sencilla la información actualizada y ordenada de los animales y productos existentes en la granja.

Pregunta N° 9: Consistencia en el diseño del sitio web en todas las secciones

Tabla 13-3: Consistencia en el diseño del sitio web en todas las secciones

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	5	0	0	0	0	5
Total usuarios	6	0	0	0	0	6
<i>Total usuarios x Total valor nivel</i>	24	0	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(6(4)+0(3)+0(2)+0(1)+0(0))*100\%}{24} = 100\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la consistencia en el diseño, se obtuvo que el 100% manifestó que si existía consistencia en el diseño del sitio web en todas las secciones del sistema; dado que los elementos (menús, submenús, base de datos, botones de comando, etc.) del sistema están bien integrados haciendo que la navegación para el usuario sea fácil en todas las secciones.

Pregunta N° 10: Combinación de colores en el diseño del sistema

Tabla 14-3: Combinación de colores en el diseño del sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Administrador	1	0	0	0	0	1
Empleados	2	2	1	0	0	5
Total usuarios	3	2	1	0	0	6
<i>Total usuarios x Total valor nivel</i>	12	6	2	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad del indicador} = \frac{(3(4)+2(3)+1(2)+0(1)+0(0))*100\%}{24} = 83.33\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la combinación de colores en el diseño del sistema, se obtuvo que el 83.33% manifestó que si es agradable la combinación de colores en el diseño del sistema informático; puesto que, el sistema está integrado con colores llamativos y suaves, que se asocian a la actividad de la empresa y a los requerimientos del usuario.

Para encontrar el porcentaje de usabilidad en cada una de las sub características, se utilizó la siguiente formula:

$$\text{Porcentaje de Usabilidad por Sub característica} = \frac{(\sum_{i=0}^{n=4} TRUNi*i)*100\%}{24*NPSC}$$

TRUN= Total de Respuestas de Usuarios Nivel

i= Variable de incremento

NPSC = Número de preguntas por Sub Característica

Resultados de comprensibilidad en el sistema

En la siguiente tabla se muestra el resultado total de las preguntas 1, 2 y 3 de las entrevistas que se utilizaron para evaluar el parámetro de comprensibilidad.

Tabla 15-3: Comprensibilidad en el sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Respuestas del Administrador	3	0	0	0	0	3
Respuestas de los Empleados	10	4	1	0	0	15
Total	13	4	1	0	0	18
Total usuarios x Total valor nivel	52	12	2	0	0	

Realizado por: Karen Pardo, 2019

$$\text{Usabilidad por Sub característica} = \frac{(13(4)+4(3)+1(2)+0(1)+0(0))*100\%}{24*3} = 91.67\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la comprensibilidad en el sistema, se obtuvo que el 91.67% manifestó que es comprensible; dado que el sistema es fácil de manejar gracias a la implementación adecuada que existe en las interfaces; así como también se comprenden los elementos que integran a éstas (menús, submenús, base de datos, botones de comando, etc.); además el usuario puede visualizar su ubicación en el sistema en todo momento.

Resultados de facilidad de aprendizaje

En la siguiente tabla se muestra el resultado total de las preguntas 4 y 5 de las entrevistas que se utilizaron para evaluar el parámetro de facilidad de aprendizaje.

Tabla 16-3: Facilidad de aprendizaje

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Respuestas del Administrador	2	0	0	0	0	2
Respuestas de los Empleados	7	3	0	0	0	10
Total	9	3	0	0	0	12
Total usuarios x Total valor nivel	36	9	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad por Sub característica} = \frac{(9(4)+3(3)+0(2)+0(1)+0(0))*100\%}{24*2} = 93.75\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la facilidad de aprendizaje, se obtuvo que el 93.75% manifestó que el sistema web es fácil de aprender; dado que el usuario no presentó complejidad al entender los elementos de las interfaces, gracias a la implementación adecuada del manual de usuario, donde muestra sencillos los procedimientos para realizar los procesos de ventas, control de productos, administración y stock de los animales existentes en la granja.

Resultados de operabilidad del sistema

En la siguiente tabla se muestra el resultado total de las preguntas 6, 7 y 8 de las entrevistas que se utilizaron para evaluar el parámetro de operabilidad.

Tabla 17-3: Operabilidad del sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Respuestas del Administrador	3	0	0	0	0	3
Respuestas de los Empleados	13	2	0	0	0	15
Total	16	2	0	0	0	18
Total usuarios x Total valor nivel	64	6	0	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad por Sub característica} = \frac{(16(4)+2(3)+0(2)+0(1)+0(0))*100\%}{24*3} = 97.22\%$$

Después de haber realizado los cálculos pertinentes en cuanto a la operabilidad del sistema, se obtuvo que el 97.22% manifestó éste si es operable con facilidad; gracias a la adecuada

integración de mensajes que presentaban las interfaces cuando el usuario cometía errores al intentar ejecutar algún proceso; además las funciones se encontraban correctamente integradas, debido a la información actualizada que el sistema mostraba al usuario; así como también la sencillez de búsqueda y filtración de información en los procesos de ventas, control de productos, administración y stock de los animales existentes en la granja.

Resultados de lo atractivo del sistema

En la siguiente tabla se muestra el resultado total de las preguntas 9 y 10 de las entrevistas que se utilizaron para evaluar el parámetro de lo atractivo del sistema.

Tabla 18-3: Atractivo del sistema

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Respuestas del Administrador	2	0	0	0	0	2
Respuestas de los Empleados	7	2	1	0	0	10
Total	9	2	1	0	0	12
Total usuarios x Total valor nivel	36	6	2	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad por Sub característica} = \frac{(9(4)+2(3)+1(2)+0(1)+0(0))*100\%}{24*2} = 91.67\%$$

Después de haber realizado los cálculos pertinentes en cuanto a lo atractivo del sistema, se obtuvo que el 91.67% manifestó que éste si es atractivo; debido a la correcta combinación de colores la cual se mostró agradable para la vista de los usuarios; además éstos manifestaron que existe consistencia en el diseño de las interfaces en todas las secciones del sistema.

Para encontrar el porcentaje de usabilidad final del sistema se utilizó la siguiente fórmula:

$$\text{Porcentaje de Usabilidad Final del Sistema} = \frac{(\sum_{i=0}^{n=4} TRUNi*i)*100\%}{24*NTP}$$

TRUN= Total de Respuestas de Usuarios Nivel

i= Variable de incremento

NTP= Número Total de Preguntas

Análisis final de Usabilidad del sistema

En la **Tabla 19-3** se muestran los resultados obtenidos de las preguntas realizadas a las personas entrevistadas de la granja porcina para medir la usabilidad del sistema. Para lo cual, se sumaron

todos los resultados de los parámetros de **comprensibilidad** (preguntas 1, 2 y 3), **facilidad de aprendizaje** (preguntas 4 y 5), **operabilidad** (preguntas 6, 7 y 8) y lo **atractivo** del sistema (preguntas 9 y 10), especificando que los resultados de la pregunta 5 que indican la facilidad de aprendizaje se invirtieron, gracias a la lógica de la misma, para que de esta manera no altere el resultado final.

Se obtuvo un total de 60 respuestas las cuales ayudaron a medir la usabilidad del sistema.

Tabla 19-3: Total de preguntas para medir la usabilidad del sistema.

<i>Usuarios</i>	<i>Siempre (4)</i>	<i>Casi Siempre (3)</i>	<i>Algunas veces (2)</i>	<i>Muy pocas veces (1)</i>	<i>Nunca (0)</i>	<i>Total</i>
Respuestas del Administrador	10	0	0	0	0	10
Respuestas de los Empleados	37	11	2	0	0	50
Total Usuarios	47	11	2	0	0	60
Total usuarios x Total valor nivel	188	33	4	0	0	

Realizado por: Karen Pardo. 2019

$$\text{Usabilidad Final del Sistema} = \frac{(47(4)+11(3)+2(2)+0(1)+0(0))*100\%}{24*10} = 93.75\%$$

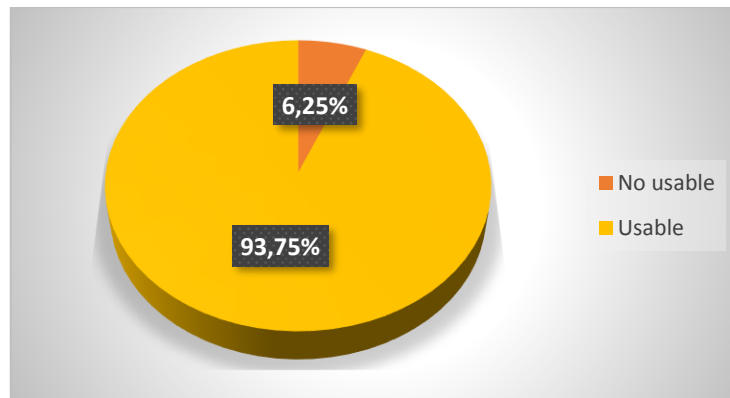


Gráfico 1-3: Usabilidad del sistema

Realizado por: Karen Pardo. 2019

Como resultado final de la evaluación de los parámetros con respecto a la usabilidad realizada a los entrevistados de la granja porcina, y basándose en las métricas de usabilidad de la norma ISO/IEC 9126 – 2, el sistema tiene un nivel de usabilidad aceptable en un 93.75%, por lo tanto, basado en la escala de usabilidad presentada en la **Tabla 4-3** se concluye que el sistema es **USABLE**.

3.1.2 Evaluación de la Eficiencia

Para medir la eficiencia del sistema, se realizaron preguntas para observar los tiempos de los procesos que tardan el administrador y los empleados de la granja porcina en forma tradicional y utilizando el sistema web.

Para obtener los resultados sobre la eficiencia del sistema, se compararon los tiempos que tardaron en realizar los procesos de manera tradicional, con los tiempos de los procesos utilizando el sistema informático.

Los resultados obtenidos se interpretan en función de la escala cualitativa representada en la siguiente tabla.

Tabla 20-3: Escala de Eficiencia

Porcentaje	Resultado
0% a 25%	Deficiente
26% a 50%	Medianamente aceptable
51% a 75%	Aceptable
76% a 100%	Eficiente

Realizado por: Karen Pardo, 2019

3.1.2.1 Medición de los Indicadores sin utilizar el sistema

En la siguiente **Tabla 21-3** se muestran los procesos para las entrevistas realizadas al administrador y a los empleados de la granja porcina de manera tradicional.

Tabla 21-3: Procesos tradicionales en la granja porcina

PROCESOS	NOMBRE
Proceso 1	Revisar y reducir el stock de productos
Proceso 2	Qué animales están a la venta
Proceso 3	Información de las salas activas
Proceso 4	Ventas entre fechas
Proceso 5	Verificación de creación de salas entre fechas
Proceso 6	Asignación de animales a sus salas

Realizado por: Karen Pardo, 2019

A continuación, se presentan los resultados de las entrevistas que se realizaron al administrador y a los empleados de la granja porcina, en lo referente a los tiempos de los procesos que se realizan de manera tradicional.

En la **Tabla 22-3** se detallan los procesos con cada uno de los tiempos que tardaron el administrador y los empleados de la granja; así mismo, se muestran los promedios por proceso en minutos.

Tabla 22-3: Tiempo en los procesos tradicionales de los usuarios que administran la granja.

Usuarios	Proceso 1 (minutos)	Proceso 2 (minutos)	Proceso 3 (minutos)	Proceso 4 (minutos)	Proceso 5 (minutos)	Proceso 6 (minutos)	Total x usuario (minutos)
Administrador	15	18	16	50	50	30	179
Empleado 1	20	15	13	50	55	25	178
Empleado 2	18	18	15	52	54	28	185
Empleado 3	16	14	14	55	52	24	175
Empleado 4	17	16	19	54	50	27	183
Empleado 5	15	15	15	55	51	25	176
Tiempo promedio	16,83	16	15,33	52,67	52	26,50	179,33

Realizado por: Karen Pardo. 2019

El promedio de los tiempos de los procesos que se realizaron sin utilizar el sistema se encuentra en un rango de 16 a 52.67 minutos.

El total de los tiempos promedio estimados

Se tiene como resultado de la medición de los tiempos promedio, un total de 179.33 minutos, que se tardan en realizar los procesos de la granja sin el uso del sistema.

3.1.2.2 Medición de los Indicadores utilizando el sistema

Ya realizada la medición de los tiempos de los procesos sin utilizar el sistema, se procedió a medir los tiempos de los procesos utilizando el sistema de venta y crianza de cerdos. Para calcular estos tiempos, se utilizó un cronómetro en línea llamado CHRON ME, el cual ayuda a medir los tiempos de cada uno de los procesos que realizaron el administrador y los usuarios en tiempo real utilizando el sistema web; este cronómetro tiene la opción de descargar un documento en Excel,

el cual incluye: la hora que inicia, el tiempo que transcurre al iniciar y la fecha que se utilizó el cronómetro. Los resultados de los tiempos que se obtuvieron con el uso del sistema se encuentran en el **ANEXO G**, el cual, evalúa la eficiencia del sistema.

A continuación, en la **Tabla 23-3** se detallan los tiempos de los procesos en segundos y minutos. Tomando en cuenta los procesos de la **Tabla 21-3**.

Tabla 23-3: Tiempo en los procesos que realizan los usuarios utilizando el sistema.

Usuarios	Proceso 1 (segundos)	Proceso 2 (segundos)	Proceso 3 (segundos)	Proceso 4 (segundos)	Proceso 5 (segundos)	Proceso 6 (segundos)	Total x Usuario (segundos)
Administrador	240	180	120	150	170	180	1040
Empleado 1	180	120	180	120	130	175	905
Empleado 2	230	160	130	180	150	160	1010
Empleado 3	175	135	175	120	130	180	915
Empleado 4	235	175	120	150	170	165	1015
Empleado 5	185	125	170	120	140	170	910
Tiempo Promedio (segundos)	207.5	149.17	149.17	140	148.33	171.67	965.84
Tiempo Promedio (minutos)	3.46	2.49	2.49	2.33	2.47	2.86	16.1

Realizado por: Karen Pardo. 2019

El promedio de los tiempos de los procesos que se realizaron utilizando el sistema, se encuentran en un rango de 2.33 a 3.46 minutos.

El total de los tiempos promedio estimados

Se tiene como resultado de la medición de los tiempos promedio un total de 16.1 minutos, que se tardan en realizar los procesos de la granja con el uso del sistema web.

3.1.2.3 Comparación de los tiempos con y sin el uso del sistema

Para finalizar con el estudio de la eficiencia del sistema, se compararon los tiempos de los procesos utilizando la forma tradicional (sin utilizar el sistema) y los tiempos de los procesos utilizando el sistema informático para el control de venta y crianza de cerdos.

Tabla 24-3: Comparación de los tiempos con y sin el uso del sistema

Procesos	Nombre	Tiempo sin el sistema (minutos)	Tiempo con el sistema (minutos)
Proceso 1	Revisar y reducir el stock de productos	16.83	3.46
Proceso 2	Qué animales están a la venta	16	2.49
Proceso 3	Información de las salas activas	15.33	2.49
Proceso 4	Ventas entre fechas	52.67	2.33
Proceso 5	Verificación de creación de salas entre fechas	52	2.47
Proceso 6	Asignación de animales a sus salas	26.50	2.86
TOTAL		179.33	16.1

Realizado por: Karen Pardo. 2019

Con la siguiente fórmula se calculó el porcentaje de reducción de tiempo para cada proceso:

$$\text{Porcentaje de Reducción de tiempo} = \left(\frac{TSS - TCS}{TSS} \right) * 100\%$$

TSS = Tiempo Sin el Sistema

TCS = Tiempo Con el Sistema

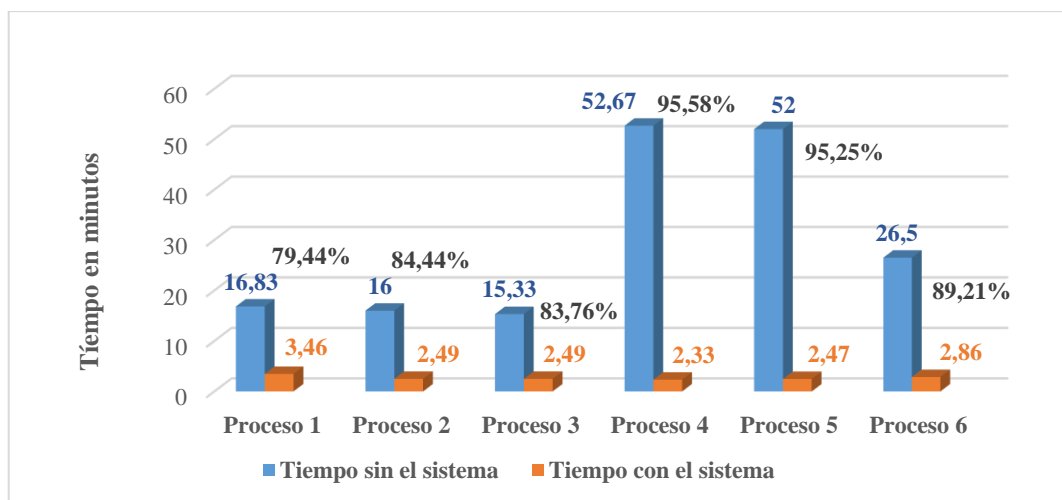


Gráfico 2-3: Comparación de los tiempos estimados

Realizado por: Karen Pardo. 2019

En el **Gráfico 2-3** se muestran los tiempos de cada uno de los procesos en minutos sin y con el uso del sistema; también se observan los porcentajes de reducción de los tiempos.

Se compararon los tiempos estimados, y se observó que el promedio del proceso 4 y 5 son los que demandan mayor tiempo sin el uso del sistema; puesto que, se trata de la verificación de ventas y verificación de creación de salas entre fechas. Sin embargo, se pudo observar que el tiempo promedio con el uso del sistema tuvo una diferencia muy notable, dado que, disminuyó de 52.67 a 2.33 minutos y de 52 a 2.47 minutos respectivamente, visto porcentualmente existe una reducción del 95.58% en el proceso 4 y un 95.25% en el proceso 5.

Análisis final sobre la Eficiencia del sistema informático

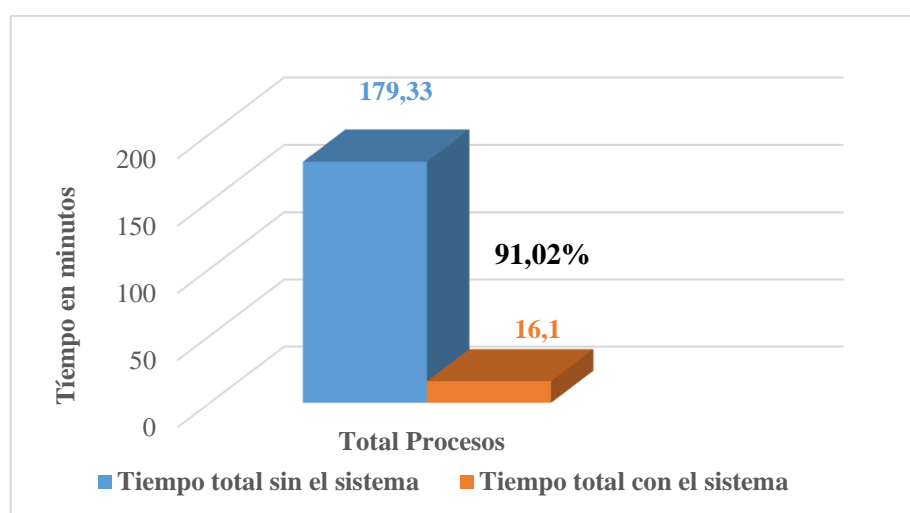


Gráfico 3-3: Comparación total de los tiempos promedio estimados

Realizado por: Karen Pardo. 2019

En el **Gráfico 3-3** se observa que existe una disminución representativa en el tiempo promedio total de los procesos, que realizaron el administrador y los empleados de la granja porcina usando el sistema informático.

El administrador y los empleados de la granja se tardaron 2 horas, 59 minutos y 24 segundos en realizar los procesos sin el uso del sistema, es decir, haciéndolo de manera tradicional. Sin embargo, con el uso del sistema el tiempo disminuyó a 16 minutos y 6 segundos. En conclusión, existió una reducción en el tiempo de los procesos en un 91.02%, el cual, basado en la escala de eficiencia planteada en la **Tabla 20-3** se determinó que el sistema informático es *Eficiente*.

CONCLUSIONES

- Los tiempos que demandaban todos los procesos para el funcionamiento de la granja eran de 2 horas, 59 minutos y 24 segundos sin usar el sistema, y con el uso del sistema el tiempo disminuyó a 16 minutos y 6 segundos; de tal manera que existe una reducción del tiempo en un 91.02%.
- Después de la instalación del sistema web se verificó la reducción de los tiempos en los procesos 4 y 5, los cuales mostraron una diferencia de 52.67 a 2.33 minutos y de 52 a 2.47 minutos respectivamente, visto porcentualmente existe una reducción del 95.58% en el proceso 4 y un 95.25% en el proceso 5.
- Por medio del análisis de las entrevistas dirigidas al administrador y a los empleados de la granja porcina “Platanillos” basada en la Norma ISO/IEC 9126-2, el sistema web para el control de venta y crianza de cerdos es utilizable, dado que el nivel de usabilidad es aceptable en un 93.75%.
- La utilización de los Framework Primefaces y Bootstrap contribuyeron satisfactoriamente con el desarrollo del proyecto, en cuanto al diseño de las interfaces del sistema, las cuales se adecuaron a los requerimientos del cliente de forma clara y precisa; asimismo lograron que el sistema sea eficiente en un 91.02%.
- La metodología Scrum ayudó en el desarrollo del sistema, dividiendo los actores y las acciones a realizarse acorde a los requerimientos del usuario para alcanzar con el objetivo deseado. Dando entregas parciales y regulares del producto, desarrollándolo de forma incremental y evolutiva.

RECOMENDACIONES

- Se recomienda la capacitación, en el ámbito informático, del personal encargado del manejo del sistema web de la granja porcina “Platanillos”, para aprovechar al 100% la funcionalidad del mismo, y así poderles brindar a los clientes una atención más rápida y eficiente.
- Se recomienda la instalación de hardware actualizado y apropiado para el aprovechamiento del sistema, la instalación de una banda ancha de internet para la buena fluidez de información, y una intranet inalámbrica para que de movilidad a los equipos de cómputo y así poderlos utilizar en cualquiera de las áreas de la granja porcina “Platanillos”.
- Se recomienda la ampliación de la usabilidad y manejo del sistema a dispositivos móviles, para facilitar el control de los procesos de la granja porcina “Platanillos”, dado que el administrador viaja constantemente a las diferentes provincias del país; por lo anterior sería de gran utilidad el manejo de la granja desde estos dispositivos.

BIBLIOGRAFÍA

- Acosta Fiallos, L.G. & Monar Borja, H.J.**, 2016. Sistema de gestión, seguimiento y evaluación de los convenios institucionales de la ESPOCH. ESPOCH. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/6311>.
- Aliaga, A., Miani, M.**, 2007. Aportaciones - I.E.S. San Vicente [WWW Document]. Disponible en: <https://iessanvicente.com/colaboraciones/postgreSQL.pdf> (accessed 12.10.18).
- Amoedo, D.**, 2018. NetBeans 8.2, instala este IDE en tu Ubuntu 18.04 [WWW Document]. Ubuunlog. Disponible en: <https://ubunlog.com/netbeans-instala-ide-ubuntu-1804/> (accessed 12.11.18).
- Arbeláez Salazar, O. & Medina Aguirre, F.A. & Chaves Osorio, J.A.**, 2011. HERRAMIENTAS PARA EL DESARROLLO RÁPIDO DE APLICACIONES WEB. Sci. Tech. XVII. Disponible en: <http://www.redalyc.org/resumen.oa?id=84921327034>.
- Aucancela Quinteros, H.L. & Cajilema Quishpi, T.P.**, 2017. Sistema informático de seguimiento de prácticas pre profesionales de la Facultad de Informática y Electrónica. ESPOCH, Riobamba. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/9089>.
- Belmonte Fernández, O.**, 2004. Introducción al lenguaje de programación Java Gratis PDF Curso. Disponible en: <https://pdfcursos.com/62-curso-introduccion-al-lenguaje-de-programacion-java.pdf>.
- Cameron, R.**, 2016. 10 Criterias for Evaluating Web Frameworks [WWW Document]. Disponible en: <https://codecondo.com/10-criterias-for-evaluating-web-frameworks/> (accessed 12.11.18).
- Cobo Yera, A.**, 2009. Diseño y programación de bases de datos. Editorial Visión Libros, Madrid España. Disponible en: https://books.google.es/books?hl=es&lr=&id=anCDr9N-kGsC&oi=fnd&pg=PA7&dq=sistema+gestor+base+de+datos+&ots=UYFDs7usuS&sig=eqn4VONOSmXz3iIvLIK9_OxIjVc#v=onepage&q=sistema%20gestor%20base%20de%20datos&f=false.
- Cordova Jaramillo, A.M. & Villamagua Morales, F.V.**, 2015. Análisis Comparativo entre los Componentes Icefaces y Primefaces de la tecnología JSF para el Desarrollo del Sistema de Inspectores Patronales del IESS Morona Santiago. ESPOCH. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/4550>.
- Díaz, F.J. & Queiruga, C.A. & Fava, L.A.**, 2007. Struts y JavaServer Faces, cara a cara. 7. Disponible en: <http://hdl.handle.net/10915/20451>.

- Egas Clavijo, P.G.**, 2015. "PRIMEFACES CRUD GENERADOR PARA NETBEANS." Universidad Central de Quito, Quito - Ecuador. Disponible en: <http://www.dspace.uce.edu.ec/bitstream/25000/4310/1/T-UCE-0011-166.pdf>.
- Egea, M.**, 2017. InControl Porcinos: la herramienta para la gestión de una granja de cerdos [WWW Document]. Softpi. Disponible en: <https://www.softpi.com/es/software-incontrol-porcinos-cerdos/> (accessed 2.18.19).
- Ferro, S. & Sanchez, M.**, 2006. JavaServerFaces. Disponible en: <http://proyectoremar.tripod.com/Documentos/Herramientas/JavaServerFaces.pdf>.
- Fiallos, I.L. & Espoch, 2018.** REGLAMENTO DE BECAS LICENCIAS COMISIONES DE SERVICIO 2018 123. Disponible en: <http://cimogsys.esPOCH.edu.ec/idi/public/docs//downloads/Marco%20Legal%20IDI-ESPOCH%202018.pdf>.
- Figuerola, R.G. & Solís, C.J. & Cabrera, A.A.**, 2007. (PDF) METODOLOGÍAS TRADICIONALES VS. METODOLOGÍAS ÁGILES [WWW Document]. Disponible en: <http://dx.doi.org/10.13140/RG.2.1.2897.3206>
- Guzman, M.**, 2012. Java - ¿Que es el lenguaje de programacion Java? [WWW Document]. Disponible en: <http://cmapspublic3.ihmc.us/rid%3D1K2C7DJ4B-4LHCV2-1WYT/Java.cmap%3Frid%3D1K2C7DJ4B-4LHCV2-1WYT%26partName%3Dhtmljpeg> (accessed 12.11.18).
- Hernandez, L. Del V.**, 2016. 51. Qué es Bootstrap [WWW Document]. Disponible en: <https://programarfacil.com/podcast/51-que-es-bootstrap/> (accessed 1.18.19).
- Herrera Saltos, P.A. & Vargas Guambo, V.M.**, 2016. Implementación de un sistema de gestión vehicular con Java Server Face para el Departamento de Mantenimiento Vehicular de la ESPOCH. ESPOCH. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/6303>.
- Jiménez Chilán, D.J.**, 2017. Desarrollo de una aplicación para el servicio de encomiendas y venta de boletos, de la Cooperativa de Transporte "Ciudad de Sucua" bajo la norma de calidad ISO/IEC 9126-2. ESPOCH. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/9087>.
- Kasián, F. & Reyes, N.S.**, 2012. Búsquedas por similitud en PostgreSQL. Presented at the XVIII Congreso Argentino de Ciencias de la Computación. Disponible en: <http://hdl.handle.net/10915/23754>.
- López, J.**, 2017. PostgreSQL – Un poco de historia y características notables [WWW Document]. SQL Train. Disponible en: <https://sqltraining.wordpress.com/2017/03/28/postgresql-un-poco-de-historia-y-caracteristicas-notables/> (accessed 12.10.18).

- Mejía Broncano, M.A.**, 2016. Sistema de seguimiento a planificación de PEA para la Facultad de Informática y Electrónica de la ESPOCH. ESPOCH, Riobamba. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/6329>.
- Merkury**, 2017. Estándares de codificación - ¡Mejora tu código! Ohmyroot! Disponible en: <https://www.ohmyroot.com/buenas-practicas-legibilidad-del-codigo/> (accessed 11.15.18).
- Muñoz Armas, O.A.**, 2011. “ESTUDIO DE FACTIBILIDAD PARA LA CREACIÓN DE UNA GRANJA DE PORCINOS EN EL RECINTO SAN FRANCISCO DE CHIPE DEL CANTÓN LA MANÁ, AÑO 2010.” UNIVERSIDAD TÉCNICA DE COTOPAXI, La Maná - Ecuador. Disponible en: <http://repositorio.utC.edu.ec/bitstream/27000/986/1/T-UTC-0687.pdf>.
- Ortega Cabrera, E.E.**, 2010. “ESTUDIO DE APLICABILIDAD Y COMPARATIVO DE UN MODELO DE CALIDAD A PRODUCTOS DE SOFTWARE CON LA NORMA ISO/IEC 9126.” ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL, Guayaquil. Disponible en: <https://www.dspace.espol.edu.ec/retrieve/90580/D-83241.pdf>.
- Pech May, F. & Gomez Rodriguez, M.A. & Lara Jeronimo, S.U.**, 2011. Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces 9. Disponible en: <https://www.tamps.cinvestav.mx/~fpech/sd/files/paper001.pdf>.
- Pepe**, 2017. Principales lenguajes de programación web, ventajas y desventajas [WWW Document]. Disponible en: <https://www.registrodominiosinternet.es/2013/08/lenguajes-programacion-web-ventajas.html> (accessed 12.7.18).
- Pérez, G.**, 2016. ¿Qué lenguaje de programación aprender? [WWW Document]. Disponible en: <http://noticias.universia.es/ciencia-tecnologia/noticia/2016/03/18/1137432/lenguaje-programacion-aprender.html> (accessed 12.11.18).
- Pérez Porto, J. & Merino, M.**, 2009. Definición de lenguaje de programación — Definicion.de [WWW Document]. Definición.de. Disponible en: <https://definicion.de/lenguaje-de-programacion/> (accessed 12.7.18).
- Pilco Guachi, N.E.**, 2018. Desarrollo de una aplicación web para el control académico de la escuela de educación básica “capullitos” aplicando la norma iso/iec 9126-2 para determinar el nivel de usabilidad. ESPOCH, Riobamba. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/9274>.
- Primetek**, 2009. PrimeFaces | Ultimate UI Framework for Java EE [WWW Document]. Disponible en: <https://www.primefaces.org/#primefaces> (accessed 1.18.19).
- Pulido Granados, E.R.**, Medina García, V.H., 2008. Modelo de medición y evaluación de la usabilidad en sitios web de la banca virtual en Colombia 22. Disponible en: <http://www.scielo.org.co/pdf/inun/v12n1/v12n1a03.pdf>.

- Quingatuña Moreano, A.V. & Nogales Guerrero, S.I.,** 2016. Desarrollo de Sistema Web Basado en Tecnologías GIS para el Mapeo de Campos Electromagnéticos en la Ciudad de Riobamba. ESPOCH - Riobamba, Riobamba. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/4751>.
- Ramos, K.,** 2017. Desarrollo de un geoportal para el Instituto Nacional de Patrimonio Cultural en la ciudad de Riobamba. ESPOCH. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/9067>.
- Rios Luna, A.,** 2014. IMPLEMENTACIÓN DEL PATRÓN MVC EN APLICACIONES WEB CON JAVA MEDIANTE LA INTEGRACIÓN DE LOS FRAMEWORK HIBERNATE, SPRING Y PRIMEFACES. Universidad Autónoma del Estado de México, México. Disponible en: <http://ri.uaemex.mx/handle/20.500.11799/62542>.
- Ruiz, G.A. et.al.** 2006. Modelo de Evaluación de Calidad de Software Basado en Lógica Difusa, Aplicada a Métricas de Usabilidad de Acuerdo con la Norma ISO/IEC 9126. Rev. Av. En Sist. E Informática 3, 29. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/9090>.
- Salesforce,** 2000. Control de Ventas: Qué es yCuál su Importancia | Salesforce [WWW Document]. Salesforce.com. Disponible en: <https://www.salesforce.com/mx/products/sales-cloud/todo-sobre-ventas/control-de-ventas/> (accessed 2.19.19).
- Schwaber, K. & Sutherland, J.,** 2017. Guía de Scrum | Guías de Scrum [WWW Document]. Disponible en: <https://www.scrumguides.org/scrum-guide.html> (accessed 1.17.19).
- Silberschatz, A. & Korth, H.F. & Sudarshan, S.,** 2002. Database system concepts, 4th ed. ed. McGraw-Hill, Boston. Disponible en: https://s3.amazonaws.com/academia.edu.documents/37358813/Fundamentos_de_Bases_de_Datos.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1544465660&Signature=GDigdoAFXNLi4zCY%2BxzPQ0pMjUU%3D&response-content-disposition=inline%3B%20filename%3DFundamentos_de_Bases_de_Datos.pdf.
- Tisalema Paninboza, M.E. & Muñoz Arias, E.A.,** 2012. Estudio Comparativo de Lenguajes de Programación Gráficos para la Implementación de un Sistema Automatizado de Seguridad Mediante Comparación de Voces, Caso Práctico: Laboratorio de Automatización Industrial de la EIS. ESPOCH. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/1527>.
- Tituaña Maldonado, W.M.,** 2017. Estudio de la integración de los framework bootstrap y primefaces para el desarrollo de aplicaciones web adaptativas con java server faces Aplicativo: Sistema de control de notas, para la unidad educativa mariano Suarez Veintimilla. Universidad Técnica del Norte, Ibarra. Ecuador. Disponible en: <http://repositorio.utn.edu.ec/handle/123456789/6903>.

- Toapanta, V.**, 2018. Desarrollo de un sistema de recaudación y facturación del agua potable para la junta parroquial de Licán, aplicando test driven development. ESPOCH. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/9129>.
- Valdiviezo Serrano, P.A. & Guacho Minta, M.A.**, 2012. Análisis Comparativo de Tecnologías de Aplicaciones Web en el Entorno JSF y ADF. Caso Práctico: IESS de Riobamba - Chimborazo. ESPOCH. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/1490>.
- Veloz, V.**, 2016. Aplicación Para La Gestión De Información Del Investigador E Integración A La Plataforma Del Proyecto Monoil. (Thesis). Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Sistemas Computacionales, Guayaquil. Disponible en: <http://repositorio.ug.edu.ec/handle/redug/16547>.
- Villalobos, G.M. & Sánchez, G.D.C. & Gutiérrez, D.A.B.**, 2010. Diseño de framework web para el desarrollo dinámico de Aplicaciones. Sci. Tech. 1, 178–183. <https://doi.org/10.22517/23447214.1817>.
- Viñé Lerma, E.**, 2010. Introducción a Primefaces [WWW Document]. Disponible en: <https://www.adictosaltrabajo.com/2010/06/30/introduccion-primefaces/> (accessed 12.11.18).
- Yuca García, J.C. & Tijero Fuentes, W.S.**, 2017. Análisis de los Framework Codeigniter PHP y Twitter Bootstrap para el desarrollo de aplicaciones web, caso: Sistema de gestión de nichos para el cementerio parque del reencuentro. Cusco - Peru. Disponible en: <http://repositorio.uandina.edu.pe:8080/xmlui/handle/UAC/879>.