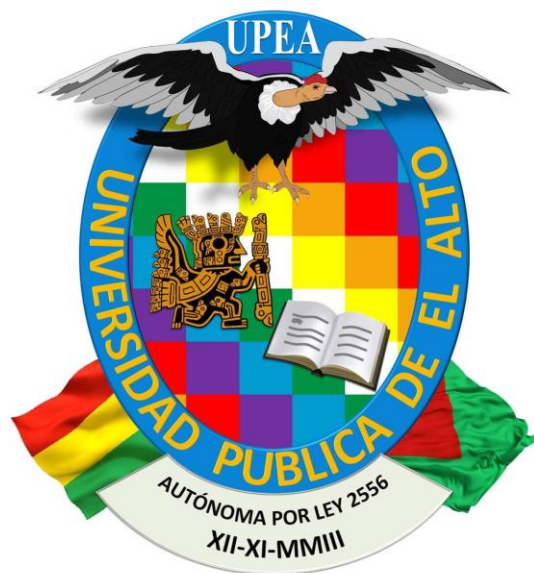


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA DE REGISTRO GEOLOCALIZACION DE VIVEROS EN EL DEPARTAMENTO DE LA PAZ”

CASO: AUTORIDAD DE FISCALIZACIÓN Y CONTROL SOCIAL DE BOSQUES Y TIERRAS (ABT)

Para Optar Al Título De Licenciatura en Ingeniería De Sistemas

MENCION: INFORMATICA Y COMUNICACIONES

POSTULANTE:	Univ. Félix Zenón Limachi Mamani
TUTOR METODOLOGICO:	Ing. Marisol Arguedas Balladares
TUTOR ESPECIALISTA:	Ing. Fanny Helen Pérez Mamani
TUTOR REVISOR:	Ing. Enrique Flores Baltazar

El Alto – Bolivia

2020

DEDICATORIA

Dedico el presente proyecto a Dios por ser el inspirador y permitirme llegar hasta estas instancias paso a pasos dados en mí convivir diario hasta llegar a este momento de mi vida.

A mis padres por ser los guía, por ser el pilar fundamental en todo lo que soy, en toda mi educación, como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo.

A mis hermanas y hermano, que me acompañaron es toda esta vida de universitario y acompañarme en las buenas y en las malas situaciones para seguir adelante, y seguir adelante.

A mis Tutores por guiarme e inculcarme sus conocimientos para el desarrollo y culminación del mismo gracias.

AGRADECIMIENTO

Primeramente a Dios, por permitirme llegar a este momento de mi vida tan especial, por los triunfos, los momentos más difíciles que me han enseñado a valorar cada día y brindarme siempre salud.

A mi Madre querida Isabela Mamani Aruquipa, que me apoyo en cada instante de mi vida dándome salud y educación ante todo.

A mis hermanos Javier Luis, Juan José, Diego y mis hermanas Virginia, Lidia, Ruth, Tereza que estuvieron siempre conmigo apoyándome en todo momento. A todos ellos darle las gracias por todo.

Al Ing. Marisol Arguedas Balladares docente tutor metodológico por tener toda la paciencia para atender mis dudas en cualquier momento, y la ayuda en cada parte de mi proyecto para culminación.

Al Ing. Enrique Flores Baltazar, docente tutor revisor de mi proyecto, por todos sus consejos profesionales, por su tiempo y entereza para dedicarse a la revisión, que le dieron forma al documento y por guiarme a lo largo del desarrollo de este proyecto con todas las observaciones y recomendaciones necesarias.

Al Ing. Fanny Helen Pérez Mamani, docente tutor especialista de mi proyecto, por brindarme su apoyo incondicional, por sus palabras emotivas que me dieron el aliento para seguir luchando profesionalmente y el tiempo dedicado, que me ayudaron a culminar este proyecto.

A la Universidad Pública de El Alto de la carrera de Ingeniería de Sistemas por acogerme y así culminar mis estudios superiores siendo una casa más que tuve.

A los docentes de la carrera de Ingeniería de Sistemas por haberme guiado y transmitirme sus conocimientos y bondades acerca del área de la mención.

Por ultimo a mis queridos amigos y amigas que siempre me han estado apoyando y ayudando durante mi estancia en la Universidad y en mi vida.

¡DE CORAZÓN MUCHAS GRACIAS A TODOS!

RESUMEN

Es evidente que hoy en día los avances y alcances de la tecnología sean expandido, aumentado aceleradamente su uso y generando una revolución muy importante en el mundo de la comunicación digital y el uso continuo del Internet, a causa de esto las instituciones tanto públicas como privadas han optado en la implementación de sistemas de información vía web, con el objetivo de facilitar difundir y promocionar los servicios que se ofrecen para su uso.

Debido a este avance tecnológico, es la razón que la institución vio la opción de contar con un Sistema Web, para obtener información de los viveros existentes y realizar una recopilación y procesamiento de datos, además de la centralización en una base de datos , y tener una georreferencia de ellos, de esta manera poder brindar información clara.

Siendo la ABT una institución dependiente de nuestro Gobierno y encargado del cuidado y preservación de nuestra madre tierra, tiene la necesidad de contar con un sistema de registro adecuado que le permita disponer de información adecuada de los viveros en nuestro departamento de La Paz.

Actualmente la institución hace la centralización de información de forma inadecuada en planillas Excel, Siendo esta situación una problemática identificada dentro de la Institución provocando un perjuicio y problema.

Viendo esta situación en la institución nace el proyecto titulado “**Sistema de Registro Geolocalización de Viveros en el Departamento de La Paz**”. El mismo que está desarrollado en atención a todas las necesidades, requerimientos y los objetos planteados por la Autoridad de Fiscalización y Control Social de Bosques y Tierras.

Para su desarrollo se aplicó en cada una de las actividades, la metodología UWE para el modelado del Sistemas Web, cumpliendo con sus fases: Modelo de requerimientos, modelo de contenido, modelo navegacional, modelo de presentación y modelo de procesos.

Sucesivamente el sistema se desarrolló con el lenguaje de programación PHP, con la ayuda del framework Bootstrap y Laravel.

El gestor de bases de datos MYSQL para almacenar toda la información recolectada, y para posterior uso.

Para medir la calidad de software del proyecto se basó en la norma ISO 9126 que hace referencia a la calidad de Software y se utilizó las siguientes fases: Funcionalidad, fiabilidad, usabilidad y mantenibilidad. La parte de seguridad se basó en la norma ISO/IEC-27000, además se trabajó con técnica de seguridad como es el algoritmo MD5 en el inicio y control de Usuarios.

Finalmente para determinar el costo y el beneficio del proyecto, se aplicó el modelo de costos COCOMO II.

ABSTRACT

It is evident that today the advances and scope of technology are expanded, its use increased rapidly and generating a very important revolution in the world of digital communication and the continuous use of the Internet, because of this, both public and private institutions They have chosen to implement information systems via the web, with the aim of facilitating the dissemination and promotion of the services offered for their use. Due to this technological advance, it is the reason that the institution saw the option of having a Web System, to obtain information from the existing nurseries and carry out data collection and processing, in addition to centralizing in a database, and having a georeference of them, in this way to provide clear information. As the ABT is an institution dependent on our Government and in charge of the care and preservation of our mother earth, it needs to have an adequate registration system that allows it to have adequate information on the nurseries in our department of La Paz.

Currently the institution makes the centralization of information in an inappropriate way in Excel forms, this situation being a problem identified within the Institution causing damage and problem.

Seeing this situation in the institution, the project entitled "Nursery Geolocation Registry System in the Department of La Paz" was born. The same that is developed in attention to all the needs, requirements and objects raised by the Authority for the Supervision and Social Control of Forests and Lands.

For its development, the UWE methodology for modeling the Web Systems was applied in each of the activities, complying with its phases: Requirements model, content model, navigational model, presentation model and process model. Subsequently the system was developed with the PHP programming language, with the help of the Bootstrap and Laravel framework. The MYSQL database manager to store all the information collected, and for later use.

To measure the software quality of the project, it was based on the ISO 9126 standard that refers to Software quality and the following phases were used: Functionality, reliability, usability and maintainability. The security part was based on the ISO / IEC-27000 standard, in addition we worked with a security technique such as the MD5 algorithm in the start and control of Users.

Finally, to determine the cost and benefit of the project, the COCOMO II cost model was applied.

INDICE

MARCO PRELIMINAR.....	1
1.1 INTRODUCCION.....	2
1.2 ANTECEDENTES.....	3
1.2.1 Institucionales.....	3
1.2.2 Internacional.....	5
1.2.3 Nacional.....	6
1.2.4 Loca.....	8
1.3 PLANTEAMIENTO DEL PROBLEMA.....	8
1.3.1 Problema Principal.....	8
1.3.2 Problemas Secundarios.....	9
1.3.3 Formulación de problema.....	9
1.4 OBJETIVOS.....	10
1.4.1 Objetivo General.....	10
1.4.2 Objetivos Específicos.....	10
1.5 JUSTIFICACION.....	10
1.5.1 Justificación Técnica.....	10
1.5.2 Justificación Económica.....	10
1.5.3 Justificación Social.....	11
1.6 METODOLOGIA.....	11
1.6.1 Metodología De Desarrollo UWE.....	11
1.7 ISO 9126.....	12
1.8 COCOMO II.....	13
1.9 ISO/IEC 27000.....	14
1.10 HERRAMIENTAS.....	14
1.10.1 Base de datos.....	14
1.10.2 Backend.....	15
1.10.3 Frontend.....	15
1.10.4 Frameworks.....	16
1.11 LIMITES Y ALCANCES.....	17
1.11.1 Limites.....	17
1.11.2 Alcances.....	18
1.12 APORTES.....	19
1.12.1 Practico.....	19
MARCO TEORICO.....	20
2.1 ANTECEDENTES.....	21
2.2 SISTEMAS.....	21
2.2.1 Clasificaciones básicas de sistemas generales.....	22
2.3 DATO.....	22
2.4 INFORMACION.....	23
2.5 SISTEMAS DE INFORMACION.....	24
2.6 MONITOREO.....	25
2.7 TIPOS DE SISTEMAS DE INFORMACION.....	26
2.8 VIVEROS.....	28
2.8.1 Tipos de viveros.....	28
2.8.2 Objetivo de los viveros.....	31

2.9	SISTEMAS DE INFORMACIÓN GEOGRÁFICA.....	31
2.10	GEOLOCALIZACION.....	33
2.11	COORDENADA.....	35
2.12	TIPOS DE DATOS EN UN SIG.....	36
2.12.1	Vector.....	36
2.12.2	Ráster.....	38
2.13	IMPORTANCIA DE LOS SISTEMAS DE INFORMACION GEOGRAFICA.....	40
2.14	INGENIERIA DE SOFTWARE.....	40
2.14.1	Ciclo de vida del software.....	41
2.14.2	Modelo de proceso de software.....	42
2.14.3	Modelo de desarrollo de software.....	43
2.14.4	Diferencia ingeniería de software y sistemas.....	44
2.14.5	Características de la ingeniería de software.....	44
2.15	METODOLOGIA.....	46
2.16	METODOLOGIA UWE.....	46
2.16.1	Conceptos.....	46
2.16.2	Características.....	47
2.16.3	Modelos.....	47
2.16.4	Aplicaciones.....	48
2.16.5	Fase de análisis de requisitos.....	48
2.16.5.1	Diagramas de casos de uso.....	49
2.16.5.2	Fase Conceptual.....	49
2.16.5.3	Fase De Navegación.....	50
2.16.5.4	Fase De Diseño De Presentación.....	51
2.16.5.5	Modelo De Proceso.....	52
2.16.6	Diagramas.....	53
2.16.6.1	Elementos del diagrama de secuencias.....	53
2.16.7	Diagrama De Estado.....	54
2.17	HERRAMIENTAS.....	55
2.17.1	Php.....	55
2.17.2	Base de datos.....	56
2.17.3	Servidor Web Apache.....	57
2.17.4	Servidor Web.....	57
2.17.5	Framework.....	58
2.17.6	Laravel.....	59
2.17.7	Javascript.....	61
2.18	PRUEBAS DE FUNCIONAMIENTO.....	62
2.18.1	Prueba de caja blanca.....	62
2.18.2	Prueba de caja negra.....	63
2.19	METRICAS DE CALIDAD.....	64
2.19.1	Características.....	65
2.19.1.1	Funcionalidad.....	66
2.19.1.2	Fiabilidad.....	69
2.19.1.3	Usabilidad.....	70
2.19.1.4	Eficiencia.....	71
2.19.1.5	Mantenibilidad.....	72
2.19.1.6	Portabilidad.....	73
2.19.1.7	Análisis comparativo de parámetros.....	74
2.19.2	Criterios de evaluación.....	76

2.20	COSTOS.....	76
2.20.1	Conceptos	76
2.20.2	Cómo II	77
2.20.3	Modelo Post-Arquitectónico.....	77
2.20.4	Constantes de complejidad	78
2.20.5	Factores multiplicadores de esfuerzo.....	78
2.21	SEGURIDAD DEL SISTEMA DE LA INFORMACION	80
2.21.1	ISO/IEC-27000	81
2.21.2	Encriptación AES-256.....	81
2.21.3	Seguridad de la base de datos	82
	MARCO APLICATIVO.....	83
3.1	INTRODUCCION	84
3.2	ANALISIS DE REQUERIMIENTO	84
3.2.1	Requerimientos funcionales	84
3.2.2	Requerimientos no funcionales	85
3.3	DISEÑO	86
3.3.1	Diagrama de caso de uso general	86
3.3.2	Caso de uso: iniciar sesión	87
3.3.3	Caso de uso: registro de usuarios de viveros.....	88
3.3.4	Caso de uso: registro de seguimientos	90
3.3.5	Caso de uso: registro de vivero.....	92
3.3.6	Caso de uso: registro de especies.....	94
3.3.7	Caso de uso: registro de plagas del vivero.....	95
3.3.8	Caso de uso: registro de localización del vivero	97
3.3.9	Caso de uso: registro de entidad financiera	98
3.3.10	Diagrama de paquetes.....	100
3.3.11	Diagrama de clases	101
3.3.12	Diagrama físico.....	102
3.4	MODELO.....	103
3.4.1	Modelo de navegación	103
3.4.1.1	Modelo de navegación de viveros	103
3.4.1.2	Modelo de navegación de seguimiento	104
3.4.1.3	Modelo de navegación de institución	104
3.4.1.4	Modelo de navegación de abastecimiento.....	105
3.4.1.5	Modelo de navegación de ubicación	105
3.4.1.6	Modelo de navegación de usuario	106
3.4.2	Modelo de presentación.....	106
3.4.2.1	Modelo de presentación registro del administrador	106
3.4.2.2	Modelo de presentación página principal	107
3.4.2.3	Modelo de presentación listado de viveros	107
3.4.2.4	Modelo de presentación listado de especies	108
3.4.2.5	Modelo de presentación listado de plagas.....	108
3.4.2.6	Modelo de presentación listado de seguimientos	109
3.4.2.7	Modelo de presentación listado de instituciones	109
3.4.2.8	Modelo de presentación listado de abastecimiento de agua	110
3.4.2.9	Modelo de presentación listado de abastecimiento de semilla.....	110
3.4.2.10	Modelo de presentación listado de departamentos	111
3.4.2.11	Modelo de presentación listado de usuarios	111

3.5	IMPLEMENTACION DEL SISTEMA	112
3.5.1	Interfaz de inicio de sesión	112
3.5.2	Interfaz de menú de navegación	112
3.5.3	Interfaz de estadísticas de los viveros	113
3.5.4	Interfaz de registro de viveros	114
3.5.5	Interfaz de seguimiento	115
3.5.6	Interfaz de registro de institución	117
3.5.7	Interfaz de registro de abastecimiento.....	119
3.5.8	Interfaz de departamentos	120
3.5.9	Interfaz de registro de usuario	120
3.6	METRICAS DE CALIDAD	121
3.6.1	Usabilidad	121
3.6.2	Funcionalidad.....	123
3.6.3	Confiabilidad	126
3.6.4	Mantenibilidad.....	128
3.6.5	Portabilidad.....	129
3.6.6	Resultados.....	130
3.7	COSTOS.....	130
3.7.1	Análisis de costos por el método Cócono II	131
3.7.2	Costo del software	132
	PRUEBAS Y RESULTADOS.....	135
4.1	PRUEBAS.....	135
4.1.1	Prueba de caja blanca	135
4.1.2	Prueba de caja negra	141
4.2	RESULTADOS	143
	CONCLUSIONES Y RECOMENTACIONES	145
5.1	CONCLUSIONES.....	146
5.2	RECOMENDACIONES	147

LISTA D E TABLAS

TABLA 1 PROBLEMA CAUSA – EFECTO.....	9
TABLA 2. DOMINIOS DE INFORMACIÓN DE PUNTOS DE FUNCIÓN	66
TABLA 3. FACTORES DE PONDERACIÓN	66
TABLA 4. VALORES DE AJUSTE DE LA COMPLEJIDAD	67
TABLA 5. MÉTRICA DE ADECUACIÓN	68
TABLA 6. MÉTRICA DE MADUREZ	69
TABLA 7. MÉTRICA DE ENTENDIBILIDAD.....	70
TABLA 8. MÉTRICA DE COMPORTAMIENTO EN EL TIEMPO.....	71
TABLA 9. MÉTRICA DE CAMBIABILIDAD	72
TABLA 10. MÉTRICA DE CONFORMIDAD DE TRANSPORTABILIDAD.....	73
TABLA 11. CONSTANTES DE COMPLEJIDAD	78
TABLA 12. VARIABLES FACTOR DE AJUSTES DEL ESFUERZO	80
TABLA 13. REPRESENTACIÓN DE REQUERIMIENTOS FUNCIONALES	85
TABLA 14. REQUERIMIENTOS NO FUNCIONALES	85
TABLA 15. DESCRIPCIÓN DEL ADMINISTRADOR	87
TABLA 16. DESCRIPCIÓN DEL REGISTRO DE USUARIOS.....	89
TABLA 17. DESCRIPCIÓN DEL REGISTRO SEGUIMIENTOS.....	91
TABLA 18. CASO DE USO REGISTRO DE VIVERO.....	92
TABLA 19. DESCRIPCIÓN DE REGISTRO DE VIVEROS.....	92
TABLA 20. DESCRIPCIÓN DE REGISTRO DE ESPECIES	94
TABLA 21. DESCRIPCIÓN DE REGISTRO DE PLAGAS.....	96
TABLA 22. DESCRIPCIÓN DE REGISTRO DE LOCALIZACIÓN	97
TABLA 23. DESCRIPCIÓN DE REGISTRO ENTIDAD FINANCIERA	99
TABLA 24. CUADRO DE USABILIDAD	122
TABLA 25. CUADRO DE PUNTO FUSIÓN Y SUS RESPECTIVAS CARACTERÍSTICAS Y PESOS.....	124
TABLA 26. CUADRO DE AJUSTE DE COMPLEJIDAD (SUMATORIAS $\sum FI$)	125
TABLA 27. CUADRO DE PORTABILIDAD	129
TABLA 28. CUADRO DE RESULTADOS	130
TABLA 29. ESTIMACIÓN DE COSTOS DEL SOFTWARE	130
TABLA 30. CONVERSIÓN DE PUNTOS DE FUNCIÓN A KLDC.....	131
TABLA 31. CONSTANTES DEL COCOMO II	132
TABLA 32. VARIABLES FACTOR DE AJUSTES DEL ESFUERZO (FAE)	133
TABLA 33. TABLA CAMINO 1.....	138
TABLA 34. TABLA CAMINO 2.....	138
TABLA 35. TABLA CAMINO 3.....	139
TABLA 36. TABLA CAMINO 4.....	139
TABLA 37. TABLA CAMINO 5.....	139
TABLA 38. TABLA CAMINO 6.....	140
TABLA 39. TABLA CAMINO 7.....	140
TABLA 40. TABLA CAMINO 8.....	141
TABLA 41. PRUEBA DE CAJA NEGRA REGISTRO DE VIVEROS.....	142
TABLA 42. PRUEBA DE CAJA NEGRA ACTUALIZACIÓN DE VIVEROS	143

LISTA DE FIGURAS

FIGURA 1. ESTRUCTURA ORGÁNICA DE FUNCIONES DE LA AUTORIDAD DE FISCALIZACIÓN Y CONTROL SOCIAL DE BOSQUES Y TIERRAS	5
FIGURA 2. VIVERO TEMPORAL	29
FIGURA 3. VIVERO PERMANENTE	30
FIGURA 4. EJEMPLOS DE LAS DIFERENTES TÉCNICAS DE GENERALIZACIÓN EN CARTOGRAFÍA	32
FIGURA 5. GEORREFERENCIACIÓN DE UN PUNTO	34
FIGURA 6. SISTEMA DE COORDENADAS GEOGRÁFICAS	35
FIGURA 7. COORDENADAS	36
FIGURA 8. TIPOS DE VECTORES	37
FIGURA 9. FORMATO RÁSTER A	38
FIGURA 10. FORMATO RÁSTER B	39
FIGURA 11. CICLOS DE LARAVEL	42
FIGURA 12. ESQUEMA DE INGENIERÍA DE SOFTWARE	45
FIGURA 13. VISTA GENERAL DE MODELOS UWE	48
FIGURA 14. DIAGRAMA DE CASO DE USO	49
FIGURA 15. DIAGRAMA DE MODELO CONCEPTUAL	49
FIGURA 16. DIAGRAMA DE MODELO NAVEGACIONAL	50
FIGURA 17. ESTEREOTIPOS DEL MODELO NAVEGACIONAL	51
FIGURA 18. DIAGRAMA DE MODELO ESTRUCTURA DE NAVEGACIÓN	51
FIGURA 19. ESTEREOTIPOS DEL MODELO DE PRESENTACIÓN	52
FIGURA 20. DIAGRAMA DE MODELO DE PRESENTACIÓN	52
FIGURA 21. ELEMENTOS DEL DIAGRAMA DE SECUENCIAS	53
FIGURA 22. DIAGRAMA DE ESTADO	54
FIGURA 23. TIPOS DE FRAMEWORK	59
FIGURA 24. CARACTERÍSTICAS DE LARAVEL	60
FIGURA 25. PRUEBA DE CAJA BLANCA	62
FIGURA 26. PRUEBA DE CAJA NEGRA	64
FIGURA 27. PROCESO DE MÉTRICAS DE CALIDAD	64
FIGURA 28. CARACTERÍSTICAS DE ISO	65
FIGURA 29. SUB CARACTERÍSTICAS DE LA NORMA ISO/IEC 9126	65
FIGURA 30. TABLA DE PARÁMETROS	75
FIGURA 31. MODELO DE CALIDAD DE USO	75
FIGURA 32. PROCESOS DE EVALUACIÓN	76
FIGURA 33. CASO DE USO GENERAL	86
FIGURA 34. CASO DE USO REGISTRO DEL ADMINISTRADOR	87
FIGURA 35. CASO DE USO REGISTRO DE USUARIOS	89
FIGURA 36. CASO DE USO REGISTRO DE SEGUIMIENTOS	90
FIGURA 37. CASO DE USO REGISTRO DE ESPECIES	94
FIGURA 38. CASO DE USO REGISTRO DE PLAGAS	95
FIGURA 39. CASO DE USO REGISTRO DE LOCALIZACIÓN	97
FIGURA 40. CASO DE USO REGISTRO ENTIDAD FINANCIERA	98
FIGURA 41. DIAGRAMA DE PAQUETES DE ABT	100
FIGURA 42. DIAGRAMA DE CLASES	101
FIGURA 43. DIAGRAMA DE DISEÑO FÍSICO	102
FIGURA 44. MODELO DE NAVEGACIÓN MENÚ PRINCIPAL	103
FIGURA 45. MODELO DE NAVEGACIÓN DE VIVEROS	103
FIGURA 46. MODELO DE NAVEGACIÓN DE SEGUIMIENTO	104
FIGURA 47. MODELO DE NAVEGACIÓN DE INSTITUCIÓN	104
FIGURA 48. MODELO DE NAVEGACIÓN DE ABASTECIMIENTO	105
FIGURA 49. MODELO DE NAVEGACIÓN UBICACIÓN	105

FIGURA 50. MODELO DE NAVEGACIÓN USUARIO	106
FIGURA 51. MODELO DE PRESENTACIÓN REGISTRO DEL ADMINISTRADOR.....	106
FIGURA 52. MODELO DE PRESENTACIÓN PÁGINA PRINCIPAL.....	107
FIGURA 53. MODELO DE PRESENTACIÓN LISTADO DE VIVEROS.....	107
FIGURA 54. MODELO DE PRESENTACIÓN LISTADO DE ESPECIES.....	108
FIGURA 55. MODELO DE PRESENTACIÓN LISTADO DE PLAGAS	108
FIGURA 56. MODELO DE PRESENTACIÓN LISTADO DE SEGUIMIENTOS	109
FIGURA 57. MODELO DE PRESENTACIÓN LISTADO DE INSTITUCIONES	109
FIGURA 58. MODELO DE PRESENTACIÓN LISTADO DE ABASTECIMIENTO DE AGUA.....	110
FIGURA 59. MODELO DE PRESENTACIÓN LISTADO DE ABASTECIMIENTO DE SEMILLA.....	110
FIGURA 60. MODELO DE PRESENTACIÓN LISTADO DE DEPARTAMENTOS	111
FIGURA 61. MODELO DE PRESENTACIÓN LISTADO DE USUARIOS.....	111
FIGURA 62. INICIO DE SESIÓN	112
FIGURA 63. PANTALLA DE PRINCIPAL (MENÚ DE NAVEGACIÓN).....	113
FIGURA 64. PANTALLA DE ESTADÍSTICAS	113
FIGURA 65. REGISTRO DE VIVEROS	114
FIGURA 66. PANTALLA DEL LISTADO DE TIPO DE ESPECIES.....	114
FIGURA 67. REGISTRO DE ESPECIES.....	115
FIGURA 68. REGISTRO DE PLAGAS	115
FIGURA 69. INTERFAZ DE SEGUIMIENTO.....	116
FIGURA 70. LISTADO DE SEGUIMIENTO	116
FIGURA 71. DOCUMENTO PDF DE VIVERO.....	117
FIGURA 72. REGISTRO DE INSTITUCIÓN.....	118
FIGURA 73. LISTADO DE INSTITUCIONES	118
FIGURA 74. LISTADO Y REGISTRO DE FUENTES DE AGUA	119
FIGURA 75. LISTADO Y REGISTRO DE SEMILLAS.....	119
FIGURA 76. LISTADO DE DEPARTAMENTO	120
FIGURA 77. REGISTRO DE UN NUEVO USUARIO.....	120
FIGURA 78. LISTADO DE USUARIOS	121
FIGURA 79. PRUEBA DE CAJA BLANCA	136

CAPITULO I
MARCO PRELIMINAR

1.1 INTRODUCCION

Actualmente son evidente los cambios agigantados y acelerados en la tecnología, que han revolucionado en el manejo de la información acorde a las necesidades de una sociedad en transformación. Hoy se ve el uso de teléfonos inteligentes o también llamados Smartphones, el uso cotidiano y necesario del internet, que son tecnología utilizada por la mayoría de las personas. Hoy existen una gran cantidad de aplicaciones disponibles, descarga gratuita para diferentes plataformas, esto para la mejor la productividad de algunas empresas, microempresas u otros. Esto nos lleva a que las instituciones grandes o pequeñas cuenten con el manejo de un sistema web para y así facilitar y maximizar su producción de una empresa con el uso del mismo. Así mismo la ABT una institución encargada para la protección y conservación de la biodiversidad, ve la necesidad de contar con un sistema que registre y geo-localicé los viveros, y de esta manera constar con información necesaria de la producción de los viveros en el departamento de La Paz.

Teniendo en cuenta el avance de la tecnología, la Institución de Autoridad de Fiscalización y Control Social de Bosques y Tierras (ABT) consta con información referente a los viveros pero estas no son suficientemente claras para los propietarios de viveros, además no se cuenta con una georreferenciación por tanto se desconoce la cantidad de ellos en el la ciudad de La Paz, así mismo no se cuenta con una base de datos para guardar toda la información recolectada.

El presente proyecto que se titula Sistema de Registro y Geolocalización de Viveros en El Departamento de La Paz, es para brindar a los propietarios información adecuada de los viveros sobre su volumen de producción, especie que se utiliza, y así mismo conocer su geolocalización de estos en la ABT (Autoridad De Fiscalización Y Control Social De Bosques Y Tierras) de esta manera la institución pueda manejar esta información adecuadamente para coadyuvar en el desarrollo sustentable de los viveros, utilizar los plantines que tienen los viveros para proyectos de forestación y reforestación.

Las herramientas utilizadas para el desarrollo del sistema son, la metodología UWE-UML para modelar aplicaciones web, el Framework Laravel para aplicación php, y servidor Apache que es un servidor web HTTP de código abierto, y como el gestor de base de datos MySql,

Para la calidad de software se utiliza la norma ISO 9126 que hace referencia a la calidad de Software y sus fases: Funcionalidad, fiabilidad, usabilidad y mantenibilidad. La parte de seguridad se basó en la norma ISO/IEC-27000, se trabajó con técnica de seguridad como es el algoritmo MD5 en el inicio y control de Usuarios.

Finalmente para determinar el costo y el beneficio del proyecto, se aplicó el modelo de costos COCOMO II.

1.2 ANTECEDENTES

1.2.1 Institucionales

a) **En el Manual De Usuario Módulo Forestación Reforestación (2019) refiere lo siguiente,** El Estado Plurinacional de Bolivia está comprometido con la protección de la Madre Tierra, es así que entre otras acciones promulgó el D.S. N° 443 que crea el Programa Nacional de Forestación y Reforestación (PNFR), que tiene como propósito:

- Ampliar la contribución a la conservación de la biodiversidad, mantenimiento de procesos ecológicos, restauración de ecosistemas y cuencas, así como la disminución de los efectos del cambio climático.
- Fortalecer la concepción de manejo integral del bosque, reconociendo el valor de la función ambiental, social y económica que cumple el mismo.
- Incrementar la cobertura boscosa del país.

Asimismo, el D.S. 29123, declara de carácter estratégico y de prioridad nacional al PNFR y señala que a través del Ministerio de Medio Ambiente y Agua (MMAyA), se implementará el módulo de forestación y reforestación del Sistema de Información y Monitoreo de Bosques (SIMB), para realizar el seguimiento al cumplimiento de las metas de forestación y reforestación. (Manual de Usuario Modulo Forestacion Reforestacion, 2019)

b) **Por otra lado el Instituto Nacional De Reforestación en su documento “Plan Nacional De Reforestación” cita,** El presente trabajo de investigación es realizar el seguimiento y monitoreo sistemático a la planificación, acciones y resultados de plantaciones forestales, en cuanto a la consolidación de las áreas forestadas y reforestadas en todos sus fines. Así de esta manera facilitar información confiable y oficial a las entidades ejecutoras y a sus autoridades, para orientar su planificación y ajustes estratégicos hacia el logro de sus metas.

Además de brindar información transparente a la población boliviana en general sobre los alcances y logros. Para impulsar con éxito las plantaciones forestales es fundamental la producción de **viveros**, por esa razón es también importante realizar el seguimiento a los mismos.

Un vivero es una instalación agronómica donde se cultivan, germinan y maduran todo tipo de plantas. Los viveros cuentan con diferentes clases de infraestructuras según su tamaño y características.

Un invernadero (espacio cerrado donde se cultivan plantas a una temperatura más alta que en el exterior), un embalse (la acumulación de agua con un fin determinado), un umbráculo (lugar destinado a la siembra y protegido del sol, la lluvia y el aire), un terreno de cultivo al aire libre y un laboratorio son algunas de las secciones que puede tener un vivero. (Minag, 2005)

- **El objetivo de la ABT**

Es ejercer el gobierno, promoviendo sistemas de desarrollo integral sustentables en los bosques y tierras, respetando los derechos e identidades culturales de los pueblos y naciones que viven y trabajan en los bosques y el área rural de Bolivia, en concordancia con los objetivos del Plan nacional de desarrollo y los preceptos de la Constitución Política del Estado.

- **Misión**

Ejercer el gobierno en los bosques y tierras: protegiendo, regulando, fiscalizando y controlando las actividades humanas, promoviendo el desarrollo y manejo integral sustentable, en beneficio del pueblo boliviano.

- **Visión**

Institución pública, técnica, eficiente y transparente que gobierna en los bosques y tierras, promoviendo el desarrollo integral y sustentable, respetando derechos y las culturas diversas, coadyuvando a la construcción de la economía plural, reconocida nacional e internacionalmente.

plántulas, ni con un método de muestreo apropiado. Se revisó y adaptó un sistema de muestreo por bancales de plántulas de ciprés, La evaluación de la calidad de las plántulas en viveros forestales se remonta al año 1917, cuando aparece un primer manual de procedimientos para la producción de viveros forestales, elaborado por el Departamento de Agricultura de los Estados Unidos de América (Mexal y Landis 1990). Sin embargo, los primeros conceptos sobre calidad de las plántulas, donde se utiliza ya claramente criterios de diámetro al cuello y altura de la plántula vigentes hoy día, aparecen en los años 30 y 40 en varios manuales publicados por investigadores citados por Mexal y Landis (1990). Con coníferas, se adaptó el sistema de clasificación de plántulas en los viveros forestales del sureste de los Estados Unidos. (Sánchez, 2004)

- b) En Trabajo de Grado: Guía De Especies Vegetales Del Vivero De Docencia Y Extensión Del Nurr - Ula Trujillo Estado Trujillo – Universidad de Los Andes,** menciona, la investigación se llevó a cabo en las instalaciones del vivero de docencia y extensión del NURR, ubicado en los predios de la Villa Universitaria en el Municipio Pampanito, sector La Concepción, y en las instalaciones de la Casa Carmona, Municipio Trujillo, sector Carmona; cuyo propósito principal fue elaborar un registro o guía con la reseña de las diferentes especies vegetales que allí se propagan y desarrollan. Para este fin, se efectuó el trabajo de campo de colección y determinación botánica respectiva, revisión bibliográfica y consulta con especialistas en el área, de esta manera se lograron identificar 30 familias para un total 73 especies vegetales; 67 de ellas para uso ornamental, 4 frutales, una forestal-medicinal y una con fines conservacionista, con esta guía se brinda a los usuarios (profesores, estudiantes y miembros de la comunidad en general) información actualizada y pertinente que será de gran utilidad principalmente para quienes están más vinculados en la ciencia del agro, así como también ofrece la oportunidad de fomentar las actividades de comercialización de plantas y cuyos recursos permiten fortalecer la estructura y funcionamiento del vivero. (Nuñez, 2012)

1.2.3 Nacional

- a) Tesis: Evaluación De Seis Especies Forestales Bajo Tres Tratamientos Pre germinativos En Vivero Comunal, Sapecho – Alto Beni U.M.S.A.** refiere, los sistemas agroforestales permiten mitigar el efecto del cambio climático mediante la absorción de dióxido de carbono, también generan ingresos económicos para el agricultor con el

aprovechamiento de especies maderables muy valiosas como el Huasicucho (*Centrolobium ochroxylum*), Quina Quina (*Myroxylon balsamum*), Teca (*Tectona grandis*), Cedro Blanco (*Cedrela fissilis*), Tarara (*Platymiscium fragans*) y Mara (*Swietenia macrophylla*). De acuerdo al trabajo de investigación se determina que el tratamiento pre-germinativo estratificación durante 8 días obtuvo mejores resultados en porcentaje de germinación en las especies forestales Tarara, Mara, Cedro Blanco, correspondientemente y Huasicucho con semillas germinadas mediante el tratamiento pre-germinativo lixiviación durante 24 horas. El mayor porcentaje de emergencia se obtuvo con el tratamiento pre-germinativo estratificación durante 8 días en semillas de Tarara, Mara, Cedro Blanco y Huasicucho. El tratamiento pre-germinativo, también logró un buen promedio en las variables: altura de planta con 28.54 cm en la especie Huasicucho; diámetro de tallo alcanzando 5.70 mm en plantines de Cedro Blanco y número de hojas presentando el mejor desarrollo en Tarara logrando una media de 18 hojas. En longitud radicular, con el tratamiento pre-germinativo escarificación en agua caliente a 85°C por 3 minutos. (Hurtado, 2015)

- b) Tesis: Efecto De Dos Tipos De Sustrato Y Tratamientos Pre-Germinativos Para Malva En El Vivero Forestal De Cota Cota U.M.S.A.** cita, El presente estudio se realizó en predios del campus de la Universidad Mayor de San Andrés (U.M.S.A.) de la Facultad de Agronomía, en la zona de Cota Cota, provincia Murillo de departamento de La Paz, tuvo como objetivo principal evaluar el efecto de dos tipos de sustrato y tratamientos pre-germinativos para la propagación de malva. La evaluación cuantitativa de los factores de estudio se realizó bajo un diseño completamente al azar de parcelas divididas, con tres repeticiones por tratamiento (Reyes, 1978), obteniendo 12 unidades experimentales, los datos obtenidos fueron analizados con el programa estadístico S.A.S. (Statistical Analysis System). Las variables de respuesta de la investigación fueron: Días a la emergencia, altura de plantas, diámetro de tallo, número de hojas, longitud de raíz y porcentaje de daño en la raíz, también se tomó en cuenta el análisis físico – químico del sustrato y el porcentaje de germinación de las semillas.

De acuerdo a los resultados las semillas que tuvieron un tratamiento pre-germinativo de remojo durante 10 minutos en agua a 60°C, obtuvo un mayor porcentaje de germinación (81,40%) respecto a las demás (66,67%). (Mendez, 2013)

1.2.4 Local

a) **Proyecto de grado: Sistema Modular de Forestación y Reforestación Fase Plantación (Ministerio de Medio Ambiente y Agua) – U.P.E.A. Carrera de Ingeniería de Sistemas,**

los bosques tiene un rol trascendental por las múltiples funciones ambientales que proveen, en el balance climático global asociado a la captura y almacenamiento de carbono, regulación del régimen hídrico, provisión de agua, protección de suelos también son escenario de desarrollo integral, social, económico, cultural, espiritual, educativo y de recreación de la sociedad. Sin embargo

Actualmente se enfrenta a los efectos del cambio climático que son cada vez más visibles y menos previsible, con mayor frecuencia e intensidad de eventos extremos que se traducen en aumento de la temperatura que a su vez causa la pérdida de nevados, la alteración en los patrones de precipitación, sequías, fuertes granizadas y en otras épocas inundaciones y por tanto genera pérdidas económica, agrícolas y de ganado, así como enfermedades, escases de agua en las ciudades y poblaciones, pérdida e incluso extinción de la biodiversidad, mayor pobreza y vulnerabilidad biofísica y social entre otras consecuencias.

En el contexto Nacional mediante la ley 300 y el DS 1696, se consolida “Mecanismo Conjunto de Mitigación y Adaptación para el Manejo Integral y Sustentable de los Bosques y Madre Tierra”, promueve la conservación, protección y restauración de los sistemas de vida de la biodiversidad y las funciones ambientales, facilitando los usos más óptimos del suelo a través del desarrollo de sistemas productivos sustentables, incluyendo agropecuarios forestales, en un contexto y adaptación al cambio climático.

Dentro de este proceso se viene avanzando en la concreción del módulo de forestación y reforestación que toma como base toda la información generada por las diferentes instituciones que participan y ejecutan el Programa Nacional de Forestación y Reforestación (PNFR). (Acho, 2018)

1.3 PLANTEAMIENTO DEL PROBLEMA

1.3.1 Problema Principal

En la Autoridad de Fiscalización y Control Social de Bosques y Tierras en el departamento de La Paz no se cuenta con un registro de la totalidad de viveros y sus características, de tal manera

han producido una incertidumbre en los tipos, especies, tenencias y además la función que cumplen son diversas, y pueden ser para producir frutas, verduras, plantines y otros, también hacer notar que es muy importante el desarrollo sustentable, siendo que las áreas verdes, por ejemplo las forestales que son el pulmón de nuestra madre tierra.

En este sentido, la información es insuficiente respecto a los viveros, es necesario contar con un registro de la cantidad de su producción de cada uno de ellos y su geolocalización para una correcta toma de decisiones.

1.3.2 Problemas Secundarios

En la siguiente tabla muestra los problemas y relación causa-efecto.

Tabla 1
Problema Causa – Efecto

N°	Causa	Efecto
1	La institución no consta con viveros geolocalizados	Se desconoce con la existencia de viveros en el departamento de La Paz
2	Se evidencia que los datos están dispersos	No se puede proporcionar información confiable y oportuna.
3	No se cuenta con un sistema que permita proporcionar informes.	El cliente (dueños de viveros) desconoce los tipos de viveros que existen
4	Las búsquedas para la obtención de información son manuales haciendo que estas tareas sean demasiado lentas.	No se cuenta con una base de datos para almacenar información
5	Perdida de información en más de una ocasión por guardar información en hojas Excel.	Debido a que el registro de los viveros es de manera manual y tradicional, usando hojas, lo cual en cualquier momento pueden extraviarse.

Fuente: Elaboración Propia

1.3.3 Formulación de problema

¿Cómo el Sistema de Registro y Geolocalización de Viveros coadyuvaría en brindar una vista general y espacial, respecto a la cantidad de recursos para facilitar el procesamiento de información en la ABT del departamento de La Paz y la toma de decisiones de forma oportuna?

1.4 OBJETIVOS

1.4.1 Objetivo General

Implementar un Sistema de Registro Geolocalización de Viveros, que brinde una vista general y espacial en tiempo real de los existentes, tipos de especies, la cantidad de plantines, optimizando el procesamiento de la información en la ABT del departamento de La Paz para la toma de decisiones de forma oportuna.

1.4.2 Objetivos Específicos

Realizar el análisis de la situación de la institución.

Sistematizar la recolección de información de los viveros.

Permitir una vista general y espacial en tiempo real de la cantidad y tipo de viveros existentes para la toma de decisiones.

Centralizar la información respecto a los viveros en una base de datos para disponer de estas de forma confiable y oportuna.

Optimizar la generación de reportes, con información que pueda utilizar la ABT y los usuarios en general.

Se realizaran las pruebas pertinentes.

1.5 JUSTIFICACION

1.5.1 Justificación Técnica

La Institución de la ABT cuenta con herramientas tecnologías de hardware como los ordenadores con amplio acceso a internet, de igual manera el software desarrollado en PHP con el Framework Laravel estará en un servidor.

De tal manera, la ingeniería del software ayudará la correcta implementación de tecnologías nuevas por lo que se dará un mejor manejo de la información almacenada.

1.5.2 Justificación Económica

Al implementar el sistema se obtendrá información confiable y clara de los viveros, por lo tanto no existirá pérdida de información al brindar informes ya que esta estará en un servidor web confiable, la institución obtendrá un mejor manejo de la información almacenada. Al desarrollar el sistema con softwares libres se ahorra la inversión económica de licencias.

1.5.3 Justificación Social

El presente sistema de registro y geolocalización permite mejorar las tareas, como brindar información actualizada a los propietarios de viveros que así lo requieran ya que está enfocado a (cantidad de producción, tipo de especie, nombre y su geolocalización) en el departamento de La Paz, de forma útil.

Este proyecto se justifica socialmente porque brindara una atención mejorada, adecuada y sistematizada para los propietarios de viveros.

Por otra parte la ABT contara con sistema eficaz para guardar toda la información recolectada de los viveros.

1.6 METODOLOGIA

La ingeniería de software es una disciplina formada por un conjunto de métodos, herramientas y técnicas que se utilizan en el desarrollo de los programas informáticos (software). Esta disciplina trasciende la actividad de programación, que es el pilar fundamental a la hora de crear una aplicación.

Por tanto la ingeniería de software, incluye el análisis previo de la situación, el diseño del proyecto, el desarrollo del software, las pruebas necesarias para confirmar su correcto funcionamiento y la implementación del sistema. En ese sentido a continuación se describe la metodología empleada en el desarrollo de este proyecto.

1.6.1 Metodología De Desarrollo UWE

La Metodología UWE – UML, Es un Método de Ingeniería de Software para el desarrollo de Web basado en UML, es decir es una herramienta basada en UML pero para aplicaciones Web, esto conlleva a que cualquier diagrama UML puede ser usado, debido a que es una extensión de UML. Esta metodología nace con la finalidad de controlar el caos que han provocado en el pasado procesos de desarrollo y así proporcionar un proceso sistemático orientado a una aplicación final de calidad. UWE es una metodología muy detallada para el desarrollo de aplicaciones que tiene una definición exacta del proceso de diseño que debe ser utilizado. Este proceso presenta flujos de trabajo y puntos de control, y sus fases son las mismas a las propuestas en el Proceso Unificado de Modelado. UWE está dirigido específicamente a aplicaciones adaptativas, basándose en características de personalización como: la definición de

un modelo de usuario en función de las preferencias, conocimiento o tareas de usuario. Además UWE usa el paradigma orientado a objetos, su orientación al usuario, la definición de una meta-modelo (modelo de referencia) que da soporte al método y el grado de formalismo que alcanza debido al soporte que proporciona para la definición de restricciones sobre los modelos.

El modelo que propone UWE está compuesto por 6 fases o sub-modelos:

- **Modelo de Casos de Uso**
- **Modelo de Contenido**
- **Modelo de Usuario**
- **Modelo de estructura**
- **Modelo Abstracto**
- **Modelo de Adaptación**

Fases de la metodología UWE

- a) **Fase de requisitos:** Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipado de la interfaz de usuario.
- b) **Fase de análisis y diseño:** UWE distingue entre diseño conceptual, de modelo de usuario, de navegación, de presentación, de adaptación, de la arquitectura, en el diseño detallado de las clases y en la definición de los subsistemas e interfaces.
- c) **Fase de implementación:** UWE incluye implementación de la arquitectura, de la estructura del hiperespacio, del modelo de usuario, de la interfaz de usuario, de los mecanismos adaptativos y las tareas referentes a la integración de todas estas implementaciones. (Eduard, 2012)

1.7 ISO 9126

Es un estándar internacional para la evaluación de la calidad del software. Está reemplazado por el proyecto SQuaRE, ISO 25000:2005, el cual sigue los mismos conceptos.

El estándar está dividido en cuatro partes las cuales dirigen, realidad, métricas externas, métricas internas y calidad en las métricas de uso y expendido. El modelo de calidad establecido en la primera parte del estándar, ISO 9126-1, clasifica la calidad del software en un conjunto estructurado de características y sub-características de la siguiente manera:

- Funcionalidad
- Fiabilidad
- Usabilidad
- Eficiencia
- Mantenibilidad
- Portabilidad
- Calidad en uso

Métricas internas son aquellas que no dependen de la ejecución del software (medidas estáticas).

Métricas externas son aquellas aplicables al software en ejecución.

Este estándar proviene desde el modelo establecido en 1977 por McCall.

- Factores (especificar): Describen la visión externa del software, como es visto por los usuarios.
- Criterios (construir): Describen la visión interna del software, como es visto por el desarrollador.
- Métricas (controlar): Se definen y se usan para proveer una escala y método para la medida.

ISO 9126 distingue entre fallo y no conformidad. Un fallo es el incumplimiento de los requisitos previos, mientras que la no conformidad es el incumplimiento de los requisitos especificados.

Una distinción similar es la que se establece entre validación y verificación.

1.8 COCOMO II

Cócono II por la época de los 90 el desarrollo del software sufrió cambios, considerando los enfoques de desarrollo de prototipos:

- Desarrollo basado en componentes (herramientas)
- Base de datos
- Modelos de desarrollo en espiral
- Paradigma orientado a objetos
- Permitiendo realizar estimaciones en función del tamaño del software
- Conjunto de factores de costo y de escala.

Está compuesto por tres modelos denominados:

- Modelo composición de Aplicación

- Modelo diseño Temprano
- Modelo de Post-Arquitectura

Los tres modelos de COCOMO II se adaptan tanto a las necesidades de los diferentes sectores descritos, como al tipo y cantidad de información disponible en cada etapa del ciclo de vida de desarrollo, lo que se conoce por granularidad de la información.

Así también para una respuesta a la diversidad del mercado actual y futuro de desarrollo de Software se propone:

- Aplicaciones desarrolladas por Usuarios Finales
- Generadores de Aplicaciones
- Aplicaciones con Componentes
- Sistemas Integrados
- Infraestructura

1.9 ISO/IEC 27000

La seguridad ISO 27000 es una norma que define de qué manera se debe implantar un Sistema de Gestión de la Seguridad de la Información en una Institución ya sea, empresa u organización. Su implantación ofrece a la ventaja de proteger su información de la forma más fiable posible, en tres objetivos:

- Preservar la confidencialidad de sus datos.
- Conservar la integridad de sus datos.
- Disponibilidad de la información protegida.

(Orozco, 2018)

1.10 HERRAMIENTAS

1.10.1 Base de datos

- **MariaDB**

Es un reemplazo mejorado y directo para el servidor de bases de datos MySQL y está disponible bajo la licencia GPL v2. Es desarrollado por la comunidad MariaDB con MariaDB Fundación como su principal desarrollador.

El API y protocolo de MariaDB es compatible con los que usa MySQL, más algunas características para soportar operaciones no bloqueantes nativas y reportes de progreso. Esto implica que todos los conectores, librerías y aplicaciones que funcionan con MySQL también deberían hacerlo con MariaDB. (Salas, 2020)

1.10.2 Backend

- **Lenguaje de programación PHP**

(Personal Hypertext processor) preprocesador de hipertexto, es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. La primera versión de PHP fue creada por Rasmus Lerdorf en 1995, y fue uno de los primeros lenguajes de programación del lado del servidor, es decir: al ser ejecutado en el servidor web directamente tiene acceso a bases de datos, conexiones en red y otras tareas antes de que se envíe la página final a la cual el cliente tendrá acceso. Además, PHP tenía la peculiaridad de poder incorporar directamente un documento HTML sin la necesidad de tener que llamar a un archivo externo que procese los datos. (Arias, 2017)

- **Servidor web Apache**

Es un servidor web HTTP de código abierto, para plataformas **Unix BSD** (Berkeley Software Distribution) fue un sistema operativo derivado de Unix Universidad de California en Berkeley, **GNU/Linux**, también conocido como **Linux**, es un sistema operativo libre tipo Unix; multiplataforma, multiusuario y multitarea. El sistema es la combinación de varios proyectos, GNU y el núcleo Linux, todo su código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera etc. (Charles, Desarrollo De Aplicacion 1º Edicion, 2004)

1.10.3 Frontend

- **HTML5**

HTML5 no es una nueva versión del antiguo lenguaje de etiquetas, ni siquiera una mejora de esta ya antigua tecnología, sino un nuevo concepto para la construcción de sitios web y aplicaciones.

El limitado objetivo de HTML motivó a varias compañías a desarrollar nuevos lenguajes y programas para agregar características a la web nunca antes implementada. Estos desarrollos

iniciales crecieron hasta convertirse en populares y poderosos accesorios. Simples juegos y bromas animadas pronto se transformaron en sofisticadas aplicaciones ofreciendo nuevas experiencias que cambiaron el concepto de la web para siempre.

JavaScript era claramente el lenguaje que permitía a los desarrolladores innovar y hacer cosas que nadie había podido hacer antes en la web. En los últimos años, programadores y diseñadores web alrededor del mundo surgieron con los más increíbles trucos para superar las limitaciones de esta tecnología y sus iniciales deficiencias en portabilidad. Gracias a estas nuevas implementaciones, Javascript, HTML y CSS se convirtieron pronto en la más perfecta combinación para la necesaria evolución de la web, HTML5 que propone estándares para cada aspecto de la web y también un propósito claro para cada una de las tecnologías involucradas. El futuro de la web es prometedor y la evolución y combinación de estas tres tecnologías (HTML, CSS y Javascript) en una poderosa especificación está volviendo a Internet la plataforma líder de desarrollo. (Gauchat, 2012)

- **CSS**

El CSS (Hojas de estilo en Cascada) es un lenguaje de estilos empleado para definir la presentación, el formato y la apariencia de un documento de marcaje, sea html, xml, o cualquier otro. Comúnmente se emplea para dar formato visual a documentos html o xhtml que funcionan como espacios web, además las hojas de estilos nacen de la necesidad de diseñar la información de tal manera que podemos separar el contenido de la presentación y, así, por una misma fuente de información, generalmente definida mediante un lenguaje de marcaje, ofrecer diferentes presentaciones en función de dispositivos, servicios, contextos o aplicativos. Por lo que un mismo documento HTML, mediante diferentes hojas de estilo, puede ser presentado por pantalla, por impresora, por lectores de voz o por tabletas braille. (Collell, s.a.)

- **JAVASCRIPT**

Javascript es un lenguaje de programación interpretado (un lenguaje de tipo script). Javascript y Java son dos lenguajes de programación distintos con filosofías muy diferentes. El único punto en común es la sintaxis, ya que cuando Netscape diseñó Javascript, se inspiró en la sintaxis de Java. (Charles, Desarrollo de aplicaciones web, 2004)

1.10.4 Frameworks

- **BOOTSTRAP**

El framework Bootstrap vio la luz en el año 2011. En un principio fue desarrollado por Twitter aunque posteriormente fue liberado bajo licencia MIT. Hoy en día continúa su desarrollo en un repositorio de GitHub. Se trata de un framework que ofrece la posibilidad de crear un sitio web totalmente responsivo mediante el uso de librerías CSS. En estas librerías, nos podemos encontrar un gran número elementos ya desarrollados y listos para ser utilizados como pueden ser botones, menús, cuadros e incluso un amplio listado de tipografías.

Desde que vio la luz, Bootstrap se ha caracterizado por tratarse de una excelente herramienta para crear interfaces de usuarios limpias y totalmente adaptables a cualquier tipo de dispositivo y pantalla, independientemente de su tamaño.

Bootstrap establece Media Queries para cuatro tamaños de dispositivos diferentes, variando dependiendo del tamaño de su pantalla. Estas Media Queries permiten desarrollar en dispositivos móviles y tabletas de forma mucho más fácil. (González, 2016)

- **LARAVEL**

laravel un framework de código abierto para desarrollar aplicaciones y servicios web con **PHP 5** y **PHP 7**. Su filosofía es desarrollar código PHP de forma elegante y simple.

Laravel simplifica enormemente la interacción con varios motores de base de datos en una variedad de entornos, se permite usar SQL puro, así como también el fluido constructor de consultas integrado con la herramienta ORM (Object Relation Mapper) Eloquent. Actualmente Laravel soporta cuatro motores por defecto, sin embargo, esta es una característica expandible según se incluyan dependencias al proyecto o cambie la conectividad del backend seleccionado: MySQL, PostgreSQL, SQLite y SQL Server.

La configuración de base de datos para cada aplicación se encuentra detallada en el archivo database.php del directorio de configuración. En este archivo se encontrarán definidas las propiedades para todas las conexiones que se requiera realizar, así como la conexión que debería ser usada por defecto. (Encada, 2018)

1.11 LIMITES Y ALCANCES

1.11.1 Limites

El Sistema de Registro y geolocalización de viveros en la ABT departamento de La Paz está centrada en brindar información confiable y completa de todos ellos.

- El administrador podrá crear, modificar e inhabilitar usuarios.
- Así mismo el proyecto está diseñado para la recolección de información referente a viveros, tipos de viveros, especies, cantidad de producción los mismos.
- Sistematizar los datos recolectados de los viveros para una toma de decisiones de forma oportuna.
- Así también proporcionara la ubicación georreferenciada de ellos.
- Generará reportes confiables de los viveros y su ubicación geográfica
- No contemplara informaciones contables.
- Así también el sistema no proporcionara información de ningún tipo de personal.
- El sistema no proporciona información de inventarios.

1.11.2 Alcances

La implementación de un sistema para el registro y geolocalización de viveros tendrá los siguientes alcances:

- Módulo de gestor de usuarios.
El administrador del sistema podrá crear, modificar e inhabilitar usuarios.
- Módulo de registro de viveros.
El administrador del sistema registra viveros según información proporcionada por los dueños de viveros
El administrador registra el tipo de vivero, la especie, la cantidad de producción,
- Módulo de ubicación de los viveros.
El administrador registra la geolocalización del vivero.
El administrador puede acceder a información desde el mapa de ubicación de los viveros
- Módulo de seguimiento.
El administrador puede brindar información de cada vivero.
El administrador brinda un reporte en pdf con información completa de los viveros.
- Módulo de registro de institución.
El administrador registrara la institución con el que cuenta el vivero.
- Módulo de abastecimiento.
El administrador registra el tipo de riego con el que cuenta el vivero.
El administrador registra el tipo de semilla que utiliza el vivero.

1.12 APORTES

1.12.1 Practico

El desarrollo del presente sistema de registro de viveros permite solucionar y/o cubrir las necesidades planteados en la institución ABT del departamento de La Paz.

El sistema realiza el registro de los viveros, de manera sistematizada.

La implementación del sistema reduce el tiempo de registro de viveros.

El sistema proporciona información confiable, integra y segura, ya que asegura la veracidad de la misma.

El sistema almacenara la información en una base de datos.

Contará con una georreferenciación de todos los viveros registrados.

CAPÍTULO II
MARCO TEORICO

2.1 ANTECEDENTES

El “Sistema de monitoreo holístico e integral de bosques” Subsistema de Gestión y Desarrollo Forestal refiere, el Proyecto de “Apoyo al Mecanismo Conjunto en Acciones de Mitigación al Cambio Climático en los Bosques de Bolivia” - UNJP/BOL/045/UNJ ejecutado bajo cooperación de la Organización de las Naciones Unidas para la Agricultura y la Alimentación (FAO), desarrolló conjuntamente con el Ministerio de Medio Ambiente y Agua, el Sistema de Monitoreo Holístico e Integral de Bosques, en el que se encuentra el Módulo de Forestación y Reforestación. La consolidación de este módulo facilitará el seguimiento de las acciones y resultados del PNFR (Programa Nacional de Forestación y Reforestación).

El seguimiento y monitoreo sistemático a la planificación, acciones y resultados de plantaciones forestales, en cuanto a la consolidación de las áreas forestadas y reforestadas en todos sus fines. Así de esta manera facilitar información confiable y oficial a las entidades ejecutoras y a sus autoridades, para orientar su planificación y ajustes estratégicos hacia el logro de sus metas. Además de brindar información transparente a la población boliviana en general sobre los alcances y logros.

En ese contexto, los datos recopilados en campo, deben ser introducidas o alimentadas al Sistema de Monitoreo Holístico e Integral de Bosques en la parte de Módulo de Forestación y Reforestación, fase VIVEROS, para ello, la presente Guía orientará a los técnicos involucrados, los procedimientos de ingreso de datos al sistema. (Manual De Usuario Modulo Forestacion Y Reforestacion, 2020)

2.2 SISTEMAS

La Reingeniería de procesos define sistema como, La reunión o conjunto de elementos relacionados, los cuales pueden ser concepto, objetos, sujetos, o puede estructurarse de conceptos, objetos y sujetos como un sistema hombre - máquina que comprende las tres clases de elementos. En definitiva, según este autor un sistema es la unión de partes o componentes, conectados en una forma organizada. (Gigch, 2016)

Por otro lado “Administración De Recursos Humanos” refiere, Sistema un conjunto de elementos (partes u órganos componentes del sistema) dinámicamente relacionados, en interacción que desarrollan una actividad (operación o proceso del sistema) para lograr un objetivo o propósito (finalidad del sistema), operando con datos, energía o materia

(insumos o entradas de recursos necesarios para poner en marcha el sistema), unidos al ambiente que rodea el sistema (con el cual se relaciona dinámicamente), y para suministrar información, energía o materia (salidas o resultados de la actividad del sistema). (Chiavenato, 1999)

- **Sistema es un conjunto de elementos que están estrechamente relacionados entre sí, que mantienen al sistema directo o indirectamente unido.**

2.2.1 Clasificaciones básicas de sistemas generales

Introducción a los Conceptos Básicos de la Teoría General de Sistemas, en un sentido amplio, la Teoría General de Sistemas (TGS) se presenta como una forma sistemática y científica de aproximación y representación de la realidad y, al mismo tiempo, como una orientación hacia una práctica estimulante para formas de trabajo transdisciplinarias. (Marcelo Arnold, 1998)

La Teoría General De Sistemas refiere que, la Metodología General de Sistemas reúne los elementos necesarios para difundir y hacer extensiva su propia aplicación, es una metodología que permite elaborar modelos y pronosticar como se comportarán antes de su puesta en marcha mediante la aplicación de procesos de simulación, permitiendo seleccionar la mejor alternativa a la problemática analizada. Implica el empleo de la lógica soportada en una amplia base tecnológica, la cual se caracteriza por tener como ingredientes la cualificación y la objetividad, lo cual permite la manipulación de grandes volúmenes de datos y restricciones de los mismos en un tiempo prudente y a unos costos razonables, coadyuvando en esta forma a la selección de la mejor alternativa en la solución del problema objeto de estudio. Pero la Metodología General de Sistemas no se limita solamente a la utilización de la tecnología para solucionar ciertos problemas humanos, pues se requiere de un enfoque más amplio, de mayor alcance, que enfrente el problema con una visión macro, incorporando factores sociológicos y tecnológicos adecuadamente balanceadas. (Alzate, 2012)

- **Un sistema es un conjunto de elementos que están estrechamente interrelacionados entre sí para un fin común y así solucionar un problema de la vida real con procesos y métodos.**

2.3 DATO

Un dato es un símbolo, tales como las letras del alfabeto, números, movimientos de labios, puntos y rayas, señales con la mano, dibujos, etc. Lo importante es considerar que estos

símbolos se pueden ordenar y reordenar de forma utilizable y se les denomina información. Los datos son símbolos que describen condiciones, hechos, situaciones o valores. Los datos se caracterizan por no contener ninguna información. Un dato puede significar una letra, un signo ortográfico o cualquier símbolo que represente una cantidad, una medida, una palabra o una descripción.

Son un el conjunto básico de hechos referentes a una persona, cosa o transacción. Incluyen cosas como: tamaño, cantidad, descripción, volumen, tasa, nombre o lugar. (Díaz M. C., 2019)

La Introducción a la Programación refiere que, un dato es la representación dentro de la computadora de algún aspecto de la realidad. Estos pueden para ser dados como entradas a un programa, procesados y devueltos como resultados. Si los datos son la representación de aspectos de la realidad, y dado que hay distintos aspectos de la realidad, estos deben poder reflejar esa diversidad. Para esto existen los tipos de datos. Un tipo de dato es un conjunto de valores que comparten las mismas características y operadores. Por ejemplo, un tipo de dato muy empleado y común en los lenguajes de programación es el tipo entero que representa a los números enteros, otro tipo de dato es el llamado tipo carácter que agrupa a los símbolos empleados en la escritura: las letras del abecedario, donde se distinguen las mayúsculas de las minúsculas, los símbolos que representan los números del cero al nueve y varios símbolos de puntuación (. , ; : ‘ “ ; etc.). Existen muchos otros tipos de datos y cada lenguaje de programación define los propios. (Aristides Dasso, 2014)

- **Un dato es una información ya se está el nombre de una persona, un valor numérico, un símbolo, siendo una representación de la realidad.**

2.4 INFORMACION

La información no es un dato conjunto. Es más bien una colección de hechos significativos y pertinentes, para el organismo u organización que los percibe. La definición de información **es un conjunto de datos significativos** y pertinentes que describen sucesos o entidades. A diferencia de los datos, la información tiene significado para quien la recibe siendo esta el resultado del procesamiento de los datos extraídos de una DB. (Díaz M. A., 2019)

Sistemas De Información Para La Administración refieren que, La información es **conocimientos basados en los datos** a los cuales, mediante un procesamiento, se les ha dato

significado, propósito y utilidad. La información es un conjunto de datos que se presentan de forma que es inteligible al receptor. Tiene un valor real o percibido para el usuario y se agrega a lo que ya conocía respecto a un suceso o un área de interés. Debe decir al receptor algo que no le era conocido anteriormente o que no podía ser pronosticado. En otras palabras, se agrega a su conocimiento pero debe ser relevante para la situación en la cual se aplicará (como al decidir qué acción tomar para aprovechar al máximo una oportunidad o para tratar algún problema. (Senn, 2019)

- **Según los conceptos, vemos que una información en un conjunto de datos de significancia que son utilizados para un fin común.**

2.5 SISTEMAS DE INFORMACION

La Introducción a los Sistemas de Información cita que, un Sistemas de Información es algo complejo, debido a la basta cantidad de enfoques y puntos de vista desde el cual, un concepto puede ser emitido. En ésta sección solo se pretende mostrar algunas de las definiciones de uso más común. **Un Sistemas de Información** siguiendo la definición de **Langefors**, es un **medio tecnológicamente implementado para el registro**, almacenando, y diseminando expresiones lingüísticas como también pata dibujar conclusiones de tales expresiones. Es una aceptada conceptualización considerar a un Sistemas de Información, aquél que permite recopilar, administrar y manipular un conjunto de datos que conforman la información necesaria para que los estamentos ejecutivos de una organización puedan realizar una toma de decisiones informada mente.

En forma general, es aquél **conjunto ordenado de elementos** (no necesariamente computacionales) que permiten manipular toda aquella información necesaria para implementar aspectos específicos de la toma de decisiones. Todo Sistema de Información, surge de la necesidad de información que experimenta una organización para implementar un conjunto específico de toma de decisiones. (Longatt, 2007)

Sistemas de Información, Tecnologías de la Información refiere que, **Un Sistema de Información** se puede definir como: “**Un conjunto formal de procesos** que, operando sobre una colección de datos estructurada según las necesidades de la empresa, recopilan elaboran y distribuyen la información (o parte de ella) necesaria para las operaciones de dicha empresa y para las actividades de dirección y control correspondientes (toma de

decisiones), que permitan desempeñar su actividad de acuerdo a su estrategia de negocio”. Por lo tanto en un sistema de información se puede encontrar **un conjunto de subsistemas** que incluyen hardware, software y medios de almacenamiento de datos juntos, con el fin de procesar entradas y convertirlas en salidas de información importantes en la toma de decisiones. (Barcelona, 2004)

- **El objetivo de un sistema de información es ayudar al desempeño de las actividades que desarrolla una empresa o institución, para lograr dichos objetivos y sus individuos adoptan procedimientos y prácticas de trabajo que deben resultar útiles y eficaces.**

2.6 MONITOREO

Definición ABC definición, el monitoreo ambiental es una acción que se despliega con la misión de conocer cuál es, cómo se encuentra, el estado de cosas en materia ambiental de un entorno y por tanto resulta ser una actividad de gran ayuda en lo que respecta al cuidado del medio ambiente ya que del resultado que arroje ese relevamiento que implica el monitoreo sabremos a ciencia cierta cuál es la situación concreta. Se observaran con detenimiento todos aquellos factores, contaminantes o elementos dañinos (sustancias químicas, toxinas, bacterias, virus, entre otros, presentes en un espacio determinado, ya sea un área de trabajo, una región territorial, por ejemplo. Y por otra parte también se ocupará de ofrecernos un pantallazo acerca de cuál es la situación de conservación de los recursos naturales de aquel lugar que se estudia.

Básicamente el monitoreo lo que hace es medir el grado de toxicidad presente en los mencionados espacios, si es grave, muy grave o prácticamente nulo y con esta información en mano, luego, poder implementar un plan que ataque en concreto el problema. (Ucha, Florencia, 2014)

Monitoreo y Evaluación refiere que, el monitoreo es el proceso sistemático de recolectar, analizar información para hacer el seguimiento al progreso de un programa en pos de la consecución de sus objetivos, y para guiar las decisiones de gestión. El monitoreo generalmente se dirige a los procesos en lo que respecta a cómo, cuándo y donde tienen a desarrollarse las actividades, quien las ejecuta y cuantas personas o entidades beneficia. Se la realiza una vez comenzado el programa y continúa durante todo el periodo de implementación, dependiendo de

las metas u objetivos del programa u del alcance de las actividades y estrategias diseñadas o implementadas

La importancia del monitoreo es importante para construir una base fuerte de evidencia para evaluar la amplia y diversa gama de intervenciones para abordarlas, siendo una herramienta para identificar y documentar y dar seguimiento de su proceso hacia la obtención de indicadores. (MUJERES, 2010)

- **Según los autores un monitoreo en la recolección de información del medio ambiente para determinar factores contaminantes en una región.**

2.7 TIPOS DE SISTEMAS DE INFORMACION

Introducción a los Sistemas de Información refiere, la conceptualización formal de un Sistema de Información es algo complejo en forma, debido a la basta cantidad de enfoques y puntos de vista desde el cual, un concepto puede ser emitido. En esta sección solo se pretende mostrar algunas de las definiciones de uso más común evidenciando sus características fundamentales.

- **Sistemas de Información Transaccionales**, logran la automatización de los procesos operativos dentro de una organización, son denominados Sistemas Transaccionales, ya que su función primordial consiste en procesar transacciones tales como pagos, cobros, pólizas, entradas, salidas
- **Sistemas de Información de Soporte para la Toma de Decisiones**, corresponden a los que apoyan el proceso de toma de decisiones, para tomas de decisiones producen información que sirve de apoyo de la toma de decisiones a los mandos intermedios y a la alta administración en el caso de organizaciones verticales.
- **Sistemas de Información Estratégicos**, tienen como objetivo en las organizaciones lograr ventajas competitivas, a través del uso de la tecnología de información. Ayudan a los niveles directivos a atacar y dirigir las cuestiones estratégicas y las tendencias a largo plazo dentro y en el entorno de la institución. (Francisco M. Gonzales, 2007)

En las tecnologías de la Información, Un Sistema de Información se puede definir como “un conjunto formal de procesos que, operando sobre una colección de datos estructurada según las necesidades de la empresa, recopilan elaboran y distribuyen la información (o parte de ella) necesaria para las operaciones de dicha empresa y para las actividades de dirección y control

correspondiente (toma de decisiones), que permitan desempeñar su actividad de acuerdo a su estrategia de negocio”

Existen diferentes tipos de sistemas:

- **Sistemas de procesamiento de transacciones**, Los sistemas de procesamiento de transacciones son sistemas que traspasan sistemas y que permiten que la organización interactúe con ambientes externos. Debido a que los administradores consultan los datos generados por el TPS para información al minuto acerca de lo que está pasando en sus compañías, es esencial para las operaciones diarias que estos sistemas funcionen lentamente y sin interrupción.
- **Sistemas de información gerencial**, Para poder ligar la información, los usuarios de un sistema de información gerencial, comparten una base de datos común. La base de datos guarda modelos que ayudan a los usuarios a interpretar y aplicar esos mismos datos. Los sistemas de información gerencial producen información que es utilizada en la toma de decisiones.
- **Sistema de apoyo a decisiones**, Un sistema de apoyo de decisiones es una de varias formas de establecer un sistema de información para una tarea clave administrativa o de organización; ciertamente, un sistema de apoyo de decisiones está realizado para una tarea administrativa o un problema específico y su uso se limita a dicho problema o tarea. Los sistemas de apoyo de decisiones suelen ser diseñados especialmente para servir a los administradores en cualquier nivel de la organización.
- **Sistemas expertos e inteligencia artificial**, La Inteligencia Artificial o **AI** (Artificial Intelligence) es la actividad de proveer a máquinas como las computadoras de la capacidad para exhibir conductas que se consideraría inteligentes si se observarían en seres humanos. La AI representa la aplicación más sofisticada de las computadoras, pues busca duplicar algunos tipos de razonamiento humano.
- **Sistemas de apoyo a decisiones de grupo**, sistema basado en ordenadores que apoya a grupos de personas que tienen una tarea (u objetivo) común, y que sirve como interfaz con un entorno compartido.
- **Sistemas de información a ejecutivos**, Un sistema de información que provee al ejecutivo acceso fácil a información interna y externa al negocio con el fin de dar seguimiento a los factores críticos del éxito, la información se origina tanto dentro de la

compañía como en su entorno. Generalmente se acepta que la información del entorno es especialmente importante en el nivel más alto. (Departamento de Ciencia, Ingeniería Nautiques, 2004)

- **Un Sistema de Información es un conjunto de subsistemas que constan de hardware, software, un medio de almacenamiento de datos, con el fin de procesar entradas y convertirlas en salidas de información.**

2.8 VIVEROS

El Manual de diseño y organización de viveros menciona que, un vivero es una instalación agronómica donde se cultivan, germinan y maduran todo tipo de plantas y plántulas, donde se crían diversas clases de especies vegetales, utilizando para ello los métodos de propagación de plantas conocidos. El Vivero cuenta con un conjunto de instalaciones, maquinarias, equipos, herramientas e insumos, para un funcionamiento eficiente y así lograr con ello una producción de alta calidad.

Un invernadero (espacio cerrado y cubierto de plástico, donde se cultivan plantas a una temperatura más alta que en el exterior), reservorio (espacio dedicado a la acumulación de agua para uso en el cultivo, doméstico o industrial), Área de sarán (utilizado para disminuir los rayos solares, vienen en varias dimensiones según el cultivo, también impide que el agua producto de las lluvias golpee directamente al cultivo; dentro de la estructura, la temperatura es menor que fuera, es inversa al invernadero), También, según el tamaño del vivero, puede tener un laboratorio de cultivo In-vitro y área de terminación de plantas. (Quiñones, 2015)

Tesis De Grado: Evaluación De Seis Especies Forestales bajo Tres Tratamientos Pre germinativos en Vivero Comunal, Sapecho -Alto Ben) refiere que, el vivero es una infraestructura adecuada para la producción y cuidado de plantas desde el almacenado o enraizamiento de estacas hasta el momento de trasplante al lugar definitivo. (Huratdo, 2014)

- **Un vivero es una construcción diseñada para la siembra de plantas, verduras y otros, controlando la temperatura para su cosecha hasta que sean utilizables.**

2.8.1 Tipos de viveros

En el Manual diseño y organización de viveros refiere que, de acuerdo con el tiempo que deben estar las plantas en los viveros, pueden ser permanentes o transitorios (temporales).

- **Viveros permanentes**

Son permanentes cuando se establecen por tiempo indefinido y, por lo tanto, necesitan de una infraestructura básica como invernaderos, camas de germinación, un sistema de riego, almacén, área de servicios a los trabajadores, parqueos, etc., equipos y un plan de producción y manejo. Algunos de estos viveros alcanzan dimensiones muy grandes, con altos niveles de tecnificación y altos costos para su mantenimiento y manejo iniciales.



Figura 2. Vivero temporal

Fuente: *Manual diseño y organización de viveros 2015*

- **Viveros temporales**

Los viveros son temporales o comunales, se establecen por periodos cortos, generalmente cerca de los lugares de siembra. Son viveros de apoyo, de adaptación o para la producción de material en pequeñas cantidades. Constan de estructuras sencillas y el costo de la instalación y el mantenimiento es bajo, generalmente se montan con materiales de la zona. Normalmente satisfacen la demanda de un proyecto específico, que tienen tiempo de terminación. Se establece en áreas de difícil acceso, pero están muy cercanos a las zonas donde se realizará la plantación; su producción predominante es la de plantas forestales. Generalmente se ubican en claros del bosque y trabajan por periodos cortos (de 2 a 4 años cuando mucho) e intermitentes, ya que la producción debe coincidir con la temporada de lluvias. Para su funcionamiento se requiere poca infraestructura y la inversión es baja. No

requieren de nivelación de terreno, se aprovechan las características naturales existente en el terreno. (Quiñones, 2015)

El Manual Del Protagonista Viveros y Semilleros clasifican los viveros según el objetivo y de acuerdo a la necesidad:

- **Vivero permanente**

Se construyen para la comercialización de plántulas por lo que requiere gran inversión en mano de obra e infraestructura.



Figura 3. Vivero Permanente

Fuente: *Manual De Prácticas De Viveros Forestales 2010*

- **Temporales o Comunales**

Son aquellos que se usan para corta duración según el cultivo que se va a sembrar.

Requieren poca inversión.

Se construyen con materiales disponibles en la finca.

Demandan mano de obra para su construcción en cada ciclo de cultivo.

(Instituto Nacional Tecnológico, 2016)

- **Según los manuales de viveros tenemos dos tipos de viveros, los temporales que son de pequeña dimensión, los permanentes que son de dimensiones más amplias que los temporales.**

2.8.2 Objetivo de los viveros

Creación De Un Vivero De Especies Nativas En El Municipio De Sesquile refiere que, Los viveros forestales tienen como objetivo preferente la producción de plantas de especies destinadas a la repoblación forestal, esta condición los hace en muchos aspectos diferentes de otros viveros, lo cual no quiere decir que ciertas prácticas no sean generalizadas a otro tipo de viveros, o que por lo contrario no se puedan utilizar técnicas propias de otras producciones (injerto, contenedores de gran volumen, etc.) diferentes a la obtención de plantas forestales.

Los objetivos básicos que debe cumplir un vivero son tres:

- **Satisfacer** las necesidades previstas de plantas forestales de todas aquellas especies requeridas en los programas de repoblación forestal.
- **Producir** plantas de buena calidad, es decir, que reúnan las características morfológicas y fisiológicas que las hacen idóneas para un objetivo específico.
- **Lograr** los objetivos anteriores a un costo mínimo, sin que ello comprometa la calidad de las plantas producidas. (Muñoz, 2007)

Según el Guía para el diseño y producción de un vivero forestal de pequeña escala de plantas en envase menciona, el objetivo de un vivero es el de producir la cantidad de plantas necesarias y que éstas sean buenas, fuertes y sanas, para que “prendan” cuando se las plante y crezcan bien, para cumplir con el objetivo de la plantación.

El objetivo del vivero debe cumplirse de un modo natural y orgánico, aprovechando los recursos disponibles en cada región y con el menor costo posible. (Navall, 2016)

- **El objetivo de los viveros es la producción de y lograr un objetivo determinado sean estos permanentes o temporales.**

2.9 SISTEMAS DE INFORMACIÓN GEOGRÁFICA

Un SIG (Sistemas de Información Geográfica) es un caso particular de SI en el que la información aparece georreferenciada es decir incluye su posición en el espacio utilizando, un sistema de coordenadas estandarizado resultado de una proyección cartográfica (generalmente UTM). Basado en una serie de capas de información espacial en formato digital que representan diversas variables, o bien capas que representan entidades a los que corresponden varias entradas en una base de datos enlazada, representan así una visión “cartográfica”.

Está constituido por:

- **Bases de datos espaciales**, en las que la realidad se codifica mediante unos modelos de datos específicos.
 - **Bases de datos temáticas**, cuya vinculación con la base de datos cartográfica permite asignar a cada punto, línea o área del territorio unos valores temáticos.
 - **Conjunto de programas**, que permiten manejar estas bases de datos de forma útil para diversos propósitos de investigación, docencia o gestión.
 - **Conjunto de ordenadores y periféricos de entrada y salida**, que constituyen el soporte físico del SIG. Estas incluyen tanto el programa de gestión de SIG cómo otros programas de apoyo. Debido a los requerimientos de velocidad, almacenamiento y memoria RAM de un SIG, generalmente es preferible destinar un ordenador en exclusiva a la implementación del SIG, bien sea actuando como servidor⁵⁰ o como ordenador personal.
 - **Comunidad de usuarios**, que pueda demandar información espacial.
 - **Administradores del sistema**, encargados de resolver los requerimientos de los usuarios bien utilizando las herramientas disponibles o bien produciendo nuevas herramientas.
- (Sarría S. F., 2019)

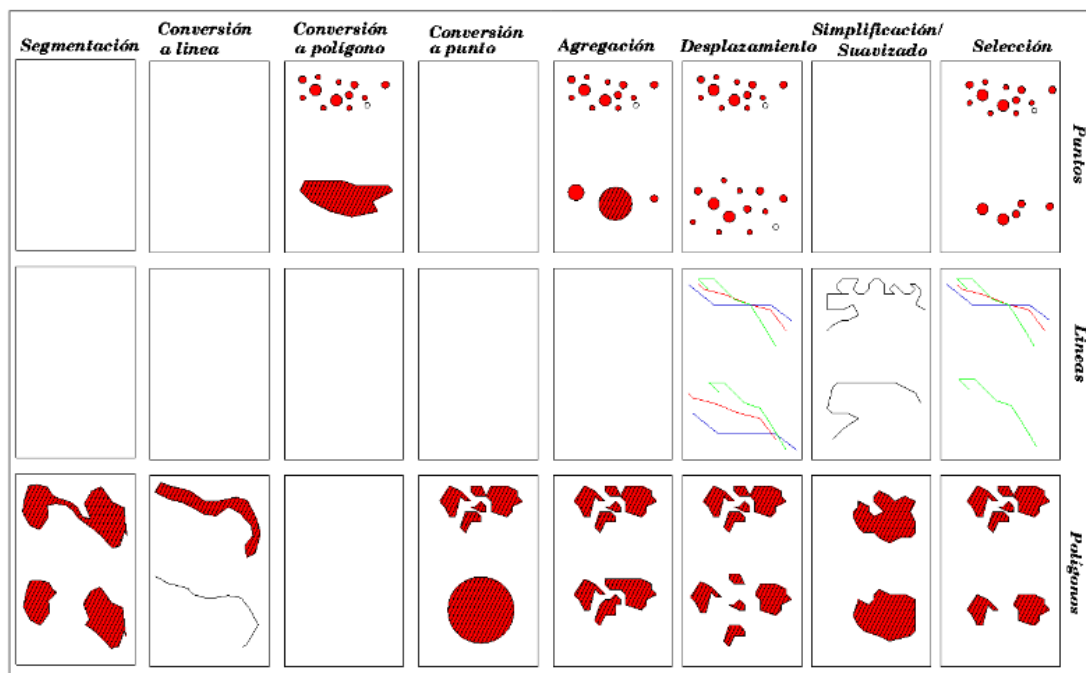


Figura 4. Ejemplos de las diferentes técnicas de generalización en cartografía
Fuente: *Sistemas de Información Geográfica 2019*

SIG describe una tecnología informática destinada al manejo de información relacionada con una ubicación geográfica. Su desarrollo y uso crece desde los años 1980 en diferentes áreas de

gestión de recursos naturales, áreas protegidas, administración urbana, pero también en aplicaciones comerciales como el mercadeo.

El concepto central de un SIG es guardar información descriptiva (atributos) junto a la ubicación referenciada con coordenadas (geometría) de elementos reales. De esta manera, los elementos reales (por ejemplo una válvula o un tramo de red) para el SIG se constituyen en objetos espaciales. Los objetos espaciales están organizados en capas, que están siendo sobrepuestas según la necesidad. (Eichholz, 2017)

- **Un sistema de información geográfica nos ayuda a representar una ubicación determinada en base a coordenadas geográficas y la utilización de capas de información espacial en formato digital.**

2.10 GEOLOCALIZACION

Influencia de un Sistema de geolocalización en el control y Monitoreo de Vehículos con Dispositivos GPS En una Empresa Logística refiere, geolocalización también denominada georreferenciación, la geolocalización implica el posicionamiento que define la localización de un objeto en un sistema de coordenadas determinado. Este proceso es generalmente empleado por los sistemas de información geográfica, un conjunto organizado de hardware y software, más datos geográficos, que se encuentra diseñado especialmente para capturar, almacenar, manipular y analizar en todas sus posibles formas la información geográfica referenciada, con la clara misión de resolver problemas de gestión y planificación.

- **Latitud**

La latitud proporciona la localización de un lugar, en dirección Norte o Sur desde el ecuador y se expresa en medidas angulares que varían desde los 0° del ecuador hasta los 90°N del polo Norte o los 90°S del polo Sur. Esto sugiere que, si trazamos una recta que vaya desde un punto cualquiera de la Tierra hasta el centro de la misma, el ángulo que forma esa recta con el plano ecuatorial expresa la latitud de dicho punto. La orientación Norte o Sur depende de si el punto marcado está por, encima del paralelo del ecuador (latitud norte) o si está por debajo de este paralelo (latitud Sur).

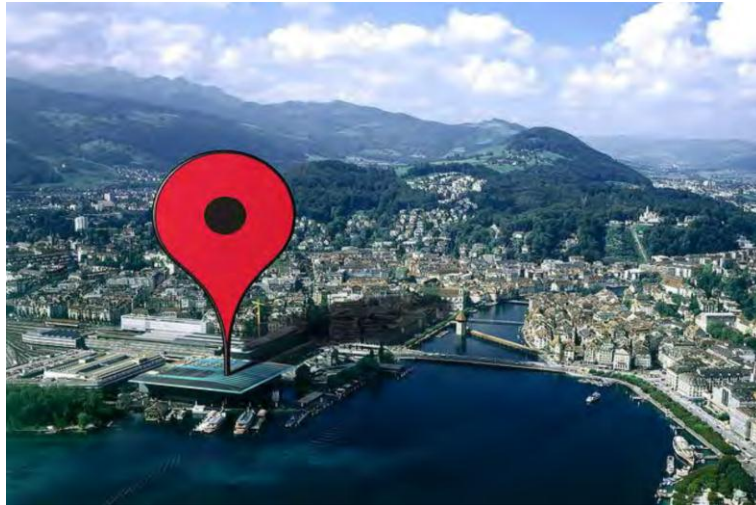


Figura 5. Georreferenciación de un punto
Fuente: *Sistemas de Información Geográfica 2019*

- **Longitud**

La longitud expresa la distancia angular entre un punto dado de la superficie terrestre y el meridiano que se tome como 0° (es decir el meridiano de base) medida a lo largo del paralelo en el que se encuentra dicho punto, una circunferencia cuyo centro es la intersección del eje de la Tierra con el plano del citado paralelo. Habitualmente en la actualidad el meridiano base es el meridiano de Greenwich (observatorio de Greenwich), pero antiguamente hubo muchos otros que servían como referencia (para el mapa de Ptolomeo el meridiano de Alejandría, para los mapas españoles hasta el siglo XIX el meridiano de Cádiz (observatorio de Cádiz) o el meridiano de Salamanca (observatorio de la Universidad de Salamanca), utilizado por la Compañía de Jesús, para los franceses el meridiano de París (observatorio de París), en Argentina a finales de siglo XIX se usó el meridiano que pasa por el antiguo observatorio de la ciudad argentina de Córdoba, etc. (Espinoza, 2017)

Por otra parte la Definición de Geolocalización” refiere que, la geolocalización es un concepto relativamente nuevo, que ha proliferado de unos años a esta parte y que hace referencia al conocimiento de la propia ubicación geográfica de modo automático. También denominada georreferenciación, la geolocalización implica el posicionamiento que define la localización de un objeto, de un dispositivo, en un sistema de coordenadas determinado de nuestro planeta tierra. Este proceso es generalmente empleado por los sistemas de información geográfica, un conjunto organizado de hardware y software, más datos geográficos, que se encuentra diseñado especialmente para capturar, almacenar, manipular y analizar en todas sus posibles formas la

información geográfica referenciada, con la clara misión de resolver problemas de gestión y planificación. Existen varias alternativas para conocer esta ubicación, aunque claro, son los dispositivos móviles los que por su portabilidad con nosotros mismos nos permitirán más fácilmente conocer nuestra ubicación y actualizarla a medida que nos vamos movilizandoy por tanto, cambiando de ubicación geográfica. (Ucha, Florencia, 2011)

- **Una geolocalización nos ayuda a la ubicación geográfica de un punto físico en un área determinada del globo terráqueo como por ejemplo el GPS de un teléfono celular.**

2.11 COORDENADA

Las coordenadas Geográficas son una forma de designar un punto sobre la superficie terrestre con el siguiente formato. El sistema de coordenadas natural de un esferoide, y por tanto de un datum (conjunto de puntos de referencia en la superficie terrestre), es el de coordenadas angulares (latitud y longitud) que suele denominarse de coordenadas geográficas. Para definir latitud y longitud, debemos identificar el eje de rotación terrestre. El plano perpendicular al eje de rotación que corta la.

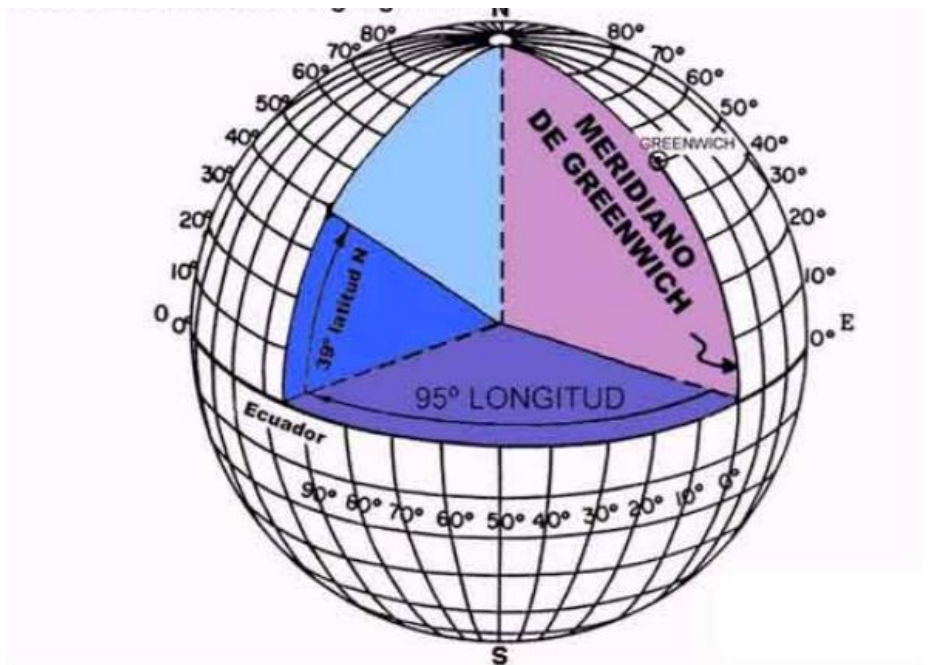


Figura 6. Sistema de coordenadas geográficas
Fuente: *Sistemas de Información Geográfica 2019*

Tierra atravesándola por su centro define el Ecuador en su intersección con el esferoide. El resto de las líneas de intersección con la superficie terrestre de los infinitos planos perpendiculares al

eje de rotación definen los diferentes paralelos o líneas de latitud constante. Finalmente, los meridianos pueden definirse como las líneas de intersección con la superficie terrestre de los infinitos planos que contienen al eje de rotación. Paralelos y meridianos se cruzan siempre en ángulo recto. (Sarria F. A., Coordenadas geográficas, 2019)

En el Sistema de Coordenadas Geográficas, un sistema que referencia cualquier punto de la superficie terrestre y que utiliza para ello dos coordenadas angulares, latitud (norte o sur) y longitud (este u oeste), para determinar los ángulos laterales de la superficie terrestre con respecto al centro de la Tierra y alineadas con su eje de rotación.

Por tanto, cualquier lugar de la superficie terrestre puede ser referenciado por la intersección de un paralelo y un meridiano, es a partir de estos conceptos de donde surgen las coordenadas de latitud y longitud. (Aristasur, 2010)

3°14'26" W
42°52'21" N

Figura 7. Coordenadas

Fuente: *Elaboración propia*

- **Un sistema de coordenadas geográficas representan un punto en cualquier punto del planeta gracias a la intersección de la longitud y latitud.**

2.12 TIPOS DE DATOS EN UN SIG

El modelo lógico hace referencia a como se muestrean y organizan las variables y entidades para lograr una representación lo más adecuada posible. En un SIG existen básicamente dos modelos lógicos que dan lugar a los dos grandes tipos de capas de información espacial:

- Formato vectorial
- Formato raster

2.12.1 Vector

En los Sistemas de Información Geográfica refiere que, los diferentes objetos se representan como puntos, líneas o polígonos. La representación de puntos o líneas es inmediata, sin embargo al representar polígonos aparecen dos situaciones diferentes.

- **Punto**

Objetos geométricos de dimensión cero, su localización espacial se representa por un par de coordenadas (X, Y), (pozos, cotas, puntos de observación, etc.).

- **Líneas**

Que simbolizan entidades, naturales o artificiales, de forma lineal (carreteras, ríos). Pueden utilizarse diferentes anchuras de línea, diferentes colores o diferentes tipos de línea para representar propiedades como la anchura de los ríos o categorías de vías de comunicación (carreteras, redes fluviales, etc.).

- **Polígonos**

Representan objetos poligonales que, por su tamaño, pueden ser representados como tales (siempre dependiendo de la escala del mapa) o porciones homogéneas del terreno en relación a una variable cualitativa (tipo de roca). Pueden utilizarse diferentes colores o tramas para representar variables cualitativas o cuantitativas, por ejemplo en un mapa de municipios se puede representar la población municipal mediante sombreados. (Entidades administrativas, ciudades, cuencas hidrográficas, etc.).

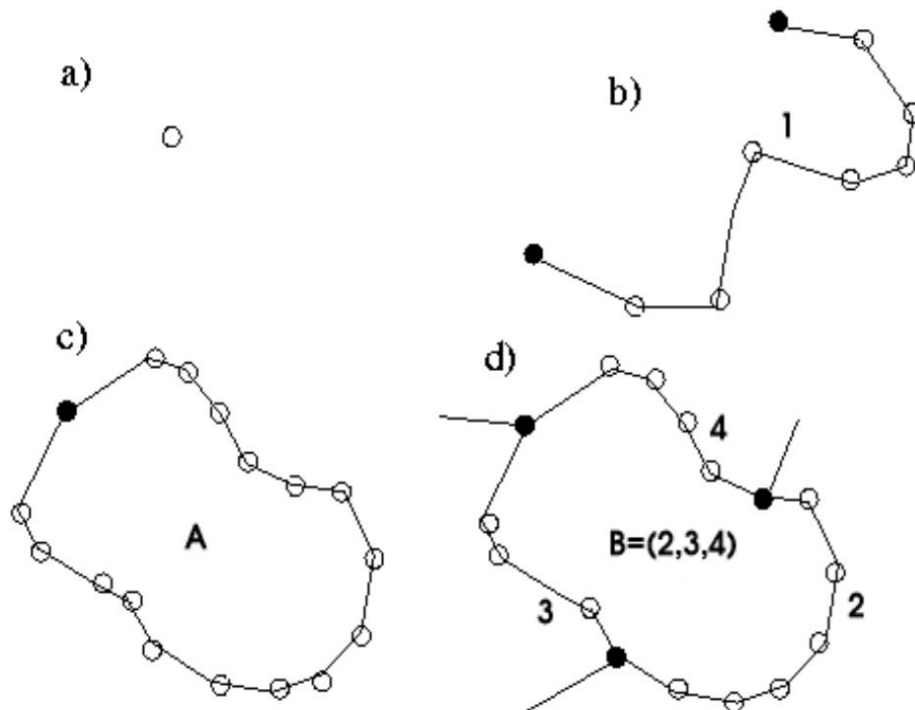


Figura 8. Tipos de vectores

Fuente: *Sistemas de Información Geográfica 2014*

2.12.2 Ráster

Un uso común de datos ráster en un SIG es en forma de visualización de fondo para otras capas de entidades. Por ejemplo las ortografías que se visualizan debajo de otra capa ofrecen al usuario de mapas la garantía de que las capas de mapas se alinean espacialmente y representan tanto objetos reales como información adicional, las tres fuentes principales de mapas base ráster son las orto-fotografías de fotografías aéreas, imágenes de satélite y mapas escaneados.

Se divide el espacio en un conjunto regular de celdillas, cada una de estas celdillas contiene un número que puede ser el identificador de un objeto (si se trata de una capa que contiene objetos) o del valor de una variable (si la capa contiene esta variable).

- **GEOTIFF**

El GeoTIFF se ha convertido en un archivo de imagen estándar en la industria de los SIG y en las aplicaciones de teledetección. Casi todos los SIG y programas de procesamiento de imágenes tienen compatibilidad con GeoTIFF.

Un GeoTIFF puede ir acompañado de otros archivos:

- **TFW** es el archivo de mundo que es requerido para dar geolocalización al raster.
- **XML** contiene los **metadatos**. Es opcional.
- **AUX** en este archivo se almacenan las proyecciones y otra información.

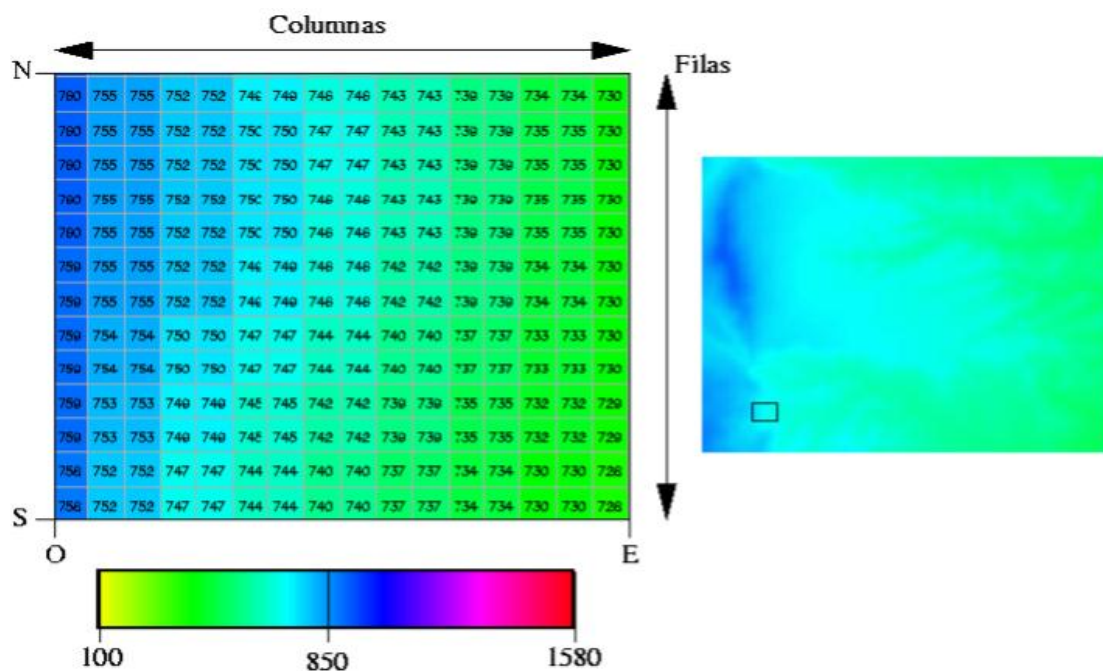


Figura 9. Formato ráster A

Fuente: *Sistemas de Información Geográfica 2014*

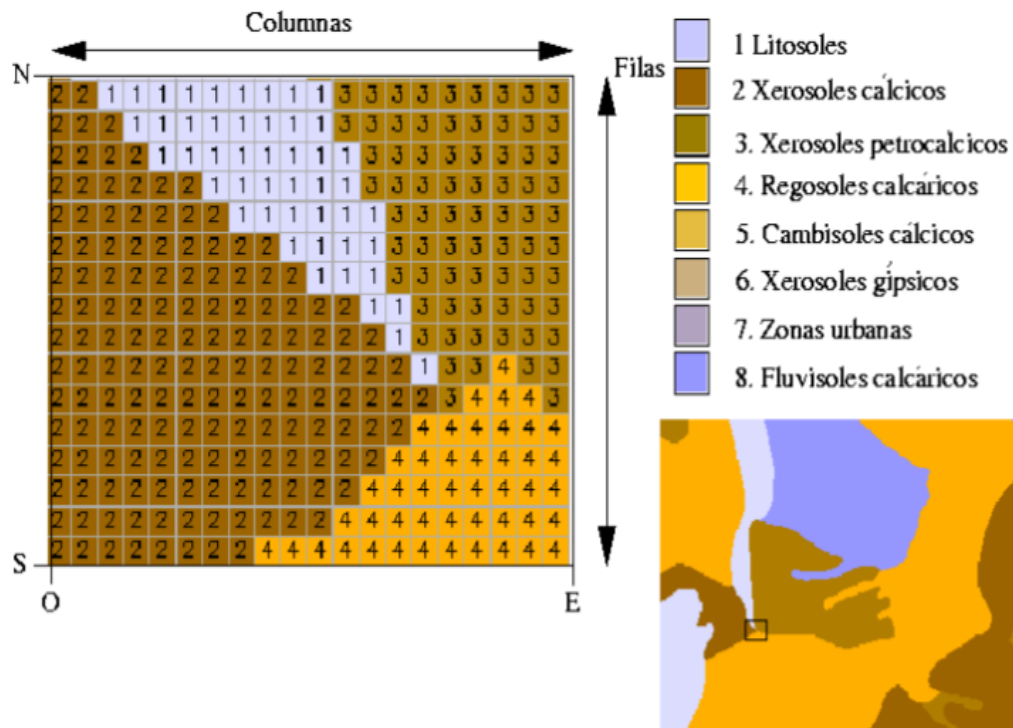


Figura 10. Formato ráster B
Fuente: *Sistemas de Información Geográfica 2014*

En origen el formato GeoTIFF fue diseñado en el Jet Propulsion Laboratory (Laboratorio de Propulsión a Reacción) de la NASA.

- **JPEG 2000**

Es un estándar de **compresión** y codificación digital de imágenes. Fue creado en el año 2000 con la intención de sustituir el formato original creado en 1992. JPEG 2000 puede trabajar con niveles de compresión mayores que los de JPEG sin incurrir en generación de bloques uniformes y aspecto borroso.

La extensión de los archivos en formato JPEG 2000 es **.jp2**. es una opción óptima para imágenes de fondo debido a la compresión sin pérdida de calidad. JPEG 2000 puede alcanzar una relación de compresión de **20:1**, similar al formato MrSID.

Los formatos de archivos ráster de **compresión de ondículas**, como JPEG 2000, ECW y MrSID, suelen tener pirámides de imágenes internas para mejorar su rendimiento y acelerar su visualización (Sarria F. A., *Sistemas de Información Geográfica*, 2014)

2.13 IMPORTANCIA DE LOS SISTEMAS DE INFORMACION GEOGRAFICA

La importancia de un (SIG) ve la necesidad de integrar el conocimiento científico a otras disciplinas, se hace necesario encontrar análisis que permitan manejar información con base espacial. Por esta razón, los Sistemas de Información Geográfica (SIG) son la herramienta idónea para estos proyectos ya que se puede utilizar información variada y de distintas fuentes que facilita el análisis simultáneo de un mismo problema.

La principal característica de un SIG es que está diseñado para trabajar con datos referenciados con respecto a coordenadas espaciales o geográficas así como trabajar con distintas bases de datos de manera integrada, permitiendo así generar información gráfica (mapas) útil para la toma de decisiones. Estos mapas ayudan a condensar varios aspectos de la realidad de una zona cuyo objetivo es reconocer la existencia de patrones espaciales sobre algún fenómeno de interés. Para generar estos mapas es necesario considerar algunos aspectos básicos:

- Análisis
- Datos fiables

La importancia del uso de un SIG en la conservación se debe a que para conocer las especies, comunidades y ecosistemas así como entender su dinámica y la importancia que tiene en la sociedad es imprescindible el conocimiento de su localización y distribución en el espacio. Una de las principales ventajas que ofrece es el poder extraer y analizar, de todo el conjunto de datos, únicamente la cantidad o variedad de datos que sea de utilidad para responder la interrogante inicial. Los SIG al trabajar de manera integrada la información permiten establecer una mejor comunicación entre varios sectores, permitiendo un balance entre las metas de los diversos sectores involucrados y facilitar la toma de decisiones que permitan el aprovechamiento de recursos de manera sostenible. (Esquivel, 2013)

2.14 INGENIERIA DE SOFTWARE

Ingeniería del Software, surge cuando se llevaba a cabo dos conferencias estructuradas por la OTAN en los años en 1967 y 1968. Es por mediados de la década los 70, cuando los sistemas informáticos incrementaron la dificultad y dan pie a la creación de las redes de computadoras. Este hecho presionó a quienes desarrollaban las computadoras personalizadas, aunque no se sabía mucho sobre ellas; al final de esta etapa, surgieron los microprocesadores.

La cuarta época del progreso de los sistemas informáticos, comienza en los 90 y se orienta hacia el impacto general de las computadoras y el software, en todos los entornos.

La Ingeniería de Software es la rama de la ingeniería que estudia todo lo relacionado con la informática o sistemas de computación, con una orientación metódica, ordenada y cuantificable al incremento, ejecución y conservación del software.

Así también la ingeniería de software es una especialidad que consiste en sistemas, instrumentos y técnicas que se emplean en el desarrollo de los programas informáticos. Que incorpora el análisis precedente de la situación, el bosquejo del proyecto, el desarrollo del software, el ensayo necesario para comprobar su funcionamiento correcto y poner en funcionamiento el sistema.

En la actualidad la ingeniería del Software es considerada una nueva especialidad de la ingeniería y junto con la especialidad de Ingeniería Informática

Se debe señalar, que el desarrollo del software va unido a lo que se conoce en el campo del software “ciclo de vida del software”

2.14.1 Ciclo de vida del software

El software no solo comprende los procesos técnicos de desarrollo de software, sino también con actividades tales como la gestión de proyectos de software y el desarrollo de herramientas, métodos y teorías de apoyo a la producción de software, con estas características es una forma de producir software de alta calidad.

Para producir un software de alta calidad el desarrollo exige un enfoque secuencial a lo largo de su vida. Abarca las siguientes actividades:

- **Análisis del Sistema:** El Software es siempre parte de un sistema mayor, por tanto se comienza estableciendo las entidades, roles, funciones etc. de los que intervienen en el sistema, se identifican los requisitos del sistema.
- **Análisis de Requisitos:** Proceso de recopilación de los requisitos específicamente del software. El analista debe comprender el ámbito de la información, la función, el rendimiento y las interfaces del software.
- **Diseño:** Traduce los requisitos en una representación de software que pueda ser codificada.
- **Codificación:** Traducción del diseño en código fuente escrito en un lenguaje de programación.

- **Prueba:** Verificación de que las funciones del software producen los resultados que realmente se requieren.
- **Mantenimiento:** El mantenimiento aplica cada uno de los pasos precedentes para implementar los cambios que con el tiempo indudablemente sufrirá el software.



Figura 11. Ciclos de Laravel
Fuente: Glaucia Mariel Atahuichi Mamani, 2014

2.14.2 Modelo de proceso de software

Los Modelos de Proceso de Software se los puede definir como una descripción simplificada de un proceso del software que presenta una visión de ese proceso. Los modelos de proceso de software, pueden incluir actividades que son parte de los procesos, productos de software y el papel de las personas involucradas en la ingeniería de del software.

La mayor parte de los modelos de procesos de software se basa en uno de los tres modelos generales o paradigmas de desarrollo de software:

- **Enfoque en cascada:** Considera las actividades anteriores y las representa con fases de procesos separados, tales como la especificación de requerimientos, el diseño de software, la implementación, las pruebas etc.

- **Desarrollo iterativo:** Este enfoque entrelaza las actividades de especificación, desarrollo y validación. Un sistema inicial se desarrolla rápidamente a partir de especificaciones muy abstractas.
- **Modelo Prototipado:** Un modelo sirve de prototipo para la construcción del sistema final.
- **Transformación Formal:** Un modelo matemático del sistema se transforma formalmente en la implementación.
- **Desarrollo basado en Reutilización:** El sistema es ensamblado a partir de componentes existentes.

2.14.3 Modelo de desarrollo de software

Una parte importante de la ingeniería de software es el desarrollo de metodologías. Donde se mencionan: Metodologías Estructurada, Metodologías Orientada a Objetos, Metodologías Tradicionales y Metodología Agiles.

- **Metodología Estructurada:** Las metodologías estructuradas se basan en la estructuración y descomposición funcional de problemas en unidades más pequeñas interrelacionadas entre sí. Representan los procesos, flujos y estructuras de datos, de una manera jerárquica y ven el sistema como entradas-proceso-salidas.
- **Metodología Orientada a Objetos:** Las metodologías orientadas a objetos modelan el sistema examinando el dominio del problema como un conjunto de objetos que interactúan entre sí.
- **Metodología Tradicional:** La metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo del software. Se centran especialmente en el control del proceso, con definición de roles, actividades, artefactos, herramientas y notaciones para el modelado y documentación detallada. Las metodologías tradicionales no se adaptan adecuadamente a los cambios.
- **Metodología Ágil:** Los procesos ágiles trabajan con requisitos desconocidos o variables. Si no existen requisitos estables, no existe una gran posibilidad de tener un diseño estable y de seguir un proceso totalmente planificado, que no vaya a variar ni en tiempo ni en dinero.

2.14.4 Diferencia ingeniería de software y sistemas

- **Ingeniería de Sistemas:** se especializa en el incremento y ejecución de las técnicas de la información. Abarca un amplio sector de trabajo, ya que toda actividad humana en la actualidad está relacionada con el manejo de información.

Los ingenieros de sistema están capacitados para ejercer cargos directivos, especialmente en el sector administrativo y también son capaces de ser líderes de una diversidad de proyectos; estudiar y desarrollar programas, asesoría e informática funcional de la organización y así servir de apoyo en las decisiones y realización de procedimientos en la operatividad.

- **Ingeniería de Software:** como ya hemos dicho, es una especialidad que consiste en sistemas, instrumentos y técnicas que se emplean en el desarrollo de los programas informáticos.

La ingeniería de software, también, incorpora el análisis precedente de la situación, el bosquejo del proyecto, el desarrollo del software, el ensayo necesario para comprobar su funcionamiento correcto y poner en funcionamiento el sistema.

Es decir, que la Ingeniería de Software es la especialidad de la ingeniería que concibe y mantiene las aplicaciones de software aplicando técnicas y pruebas de las ciencias computacionales, conducción de proyectos y otros campos.

Mientras que la Ingeniería de Sistemas es la especialidad de la ingeniería que se basa en la aplicación de los principios de las ciencias relacionadas con la computación, la electrónica y los estudios de software, para el desempeño de soluciones globales de computación y correspondencia, capaces de tratar información de manera automática.

Con lo anteriormente expuesto, llegamos a la conclusión, que ambas carreras tienen puntos en común y hasta podría decirse que se complementan y es posible recurrir a ambas al momento que se elabora un proyecto.

2.14.5 Características de la ingeniería de software

Es una especialidad de la ingeniería que tiene como objetivo principal, el desarrollo costeable de sistemas de software confiables que funcionen de modo eficiente y comprende todos los aspectos de la producción del software.

El software presenta 3 elementos que lo caracterizan:

- Los programas y/o algoritmos.
- Las estructuras de datos.
- Los documentos.

(Mamani, 2014)

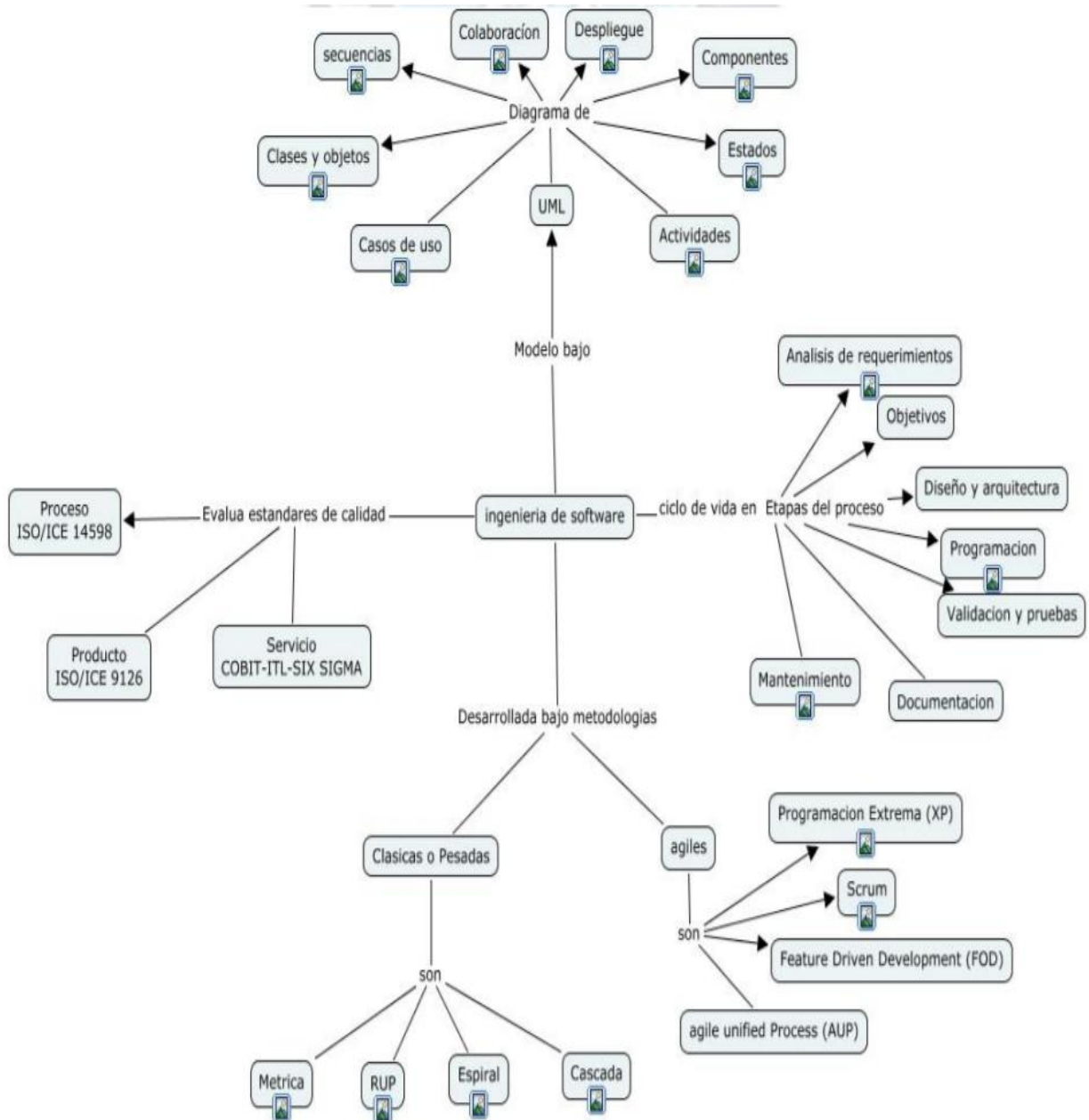


Figura 12. Esquema de Ingeniería de Software

Fuente: micarrerauniversitaria

2.15 METODOLOGIA

Metodología se denomina la serie de métodos y técnicas de rigor científico que se aplican sistemáticamente durante un proceso de investigación para alcanzar un resultado teóricamente válido. En este sentido, la metodología funciona como el soporte conceptual que rige la manera en que aplicamos los procedimientos en una investigación.

La palabra, como tal, proviene del griego (méthodos), que significa ‘método’, y el sufijo logía, que deriva de (lógos) y traduce ‘ciencia, estudio, tratado’. De allí que también sea definida como la ciencia del método.

Podemos encontrar metodología en distintas áreas de estudio, como la metodología didáctica en Educación, o la jurídica en Derecho, del mismo modo como para la solución de problemas determinados podemos aplicar una serie de pasos específicos que, en suma, funcionan como una metodología.

Metodología del desarrollo de software, hace referencia al conjunto de técnicas, procedimientos y soportes documentales empleados en el diseño de sistemas de información. Su objetivo principal es exponer una serie de técnicas clásicas y modernas de modelado de sistemas que permitan desarrollar un software de calidad, que incluyen heurísticas de construcción y criterios de comparación de modelos de sistemas.

Entre las metodologías de desarrollo de software más aplicadas en la actualidad se encuentran:

2.16 METODOLOGIA UWE

2.16.1 Conceptos

UWE es un método de ingeniería del software para el desarrollo de aplicaciones web basado en UML. Cualquier tipo de diagrama UML puede ser usado, porque UWE es una extensión de UML. Es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

Es una herramienta que nos permitirá modelar aplicaciones web, utilizada en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos). UWE es una propuesta basada en el proceso unificado y UML pero adaptados a la web. En requisitos

separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

UWE define vistas especiales representadas gráficamente por diagramas en UML. Además UWE no limita el número de vistas posibles de una aplicación, UML proporciona mecanismos de extensión basados en estereotipos. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son lo que finalmente se utilizarán en las vistas especiales para el modelado de aplicaciones Web. De esta manera, se obtiene una notación UML adecuada a un dominio en específico a la cual se le conoce como Perfil UML.

UWE está especializada en la especificación de aplicaciones adaptativas, y por tanto hace especial hincapié en características de personalización, como es la definición de un modelo de usuario o una etapa de definición de características adaptativas de la navegación en función de las preferencias, conocimiento o tareas de usuario. (Cantarero, 2016)

2.16.2 Características

Las principales características en los que se fundamenta UWE son los siguientes:

Una de las características de UWE es el uso de una notación estándar, para todos los modelos Lenguaje de modelado unificado UML.

- **Definición de métodos**

UWE presenta una definición de los pasos para la construcción de los diferentes modelos.

- **Especificación de Restricciones**

En la metodología UWE, se recomienda el uso de restricciones en su desarrollo.

2.16.3 Modelos

El proceso de UWE está dividido en cuatro actividades, que en el siguiente capítulo será profundizado con claridad, estos son los siguientes:

- Modelo de Requerimientos
- Modelo Conceptual
- Modelo de Navegación
- Modelo de Presentación

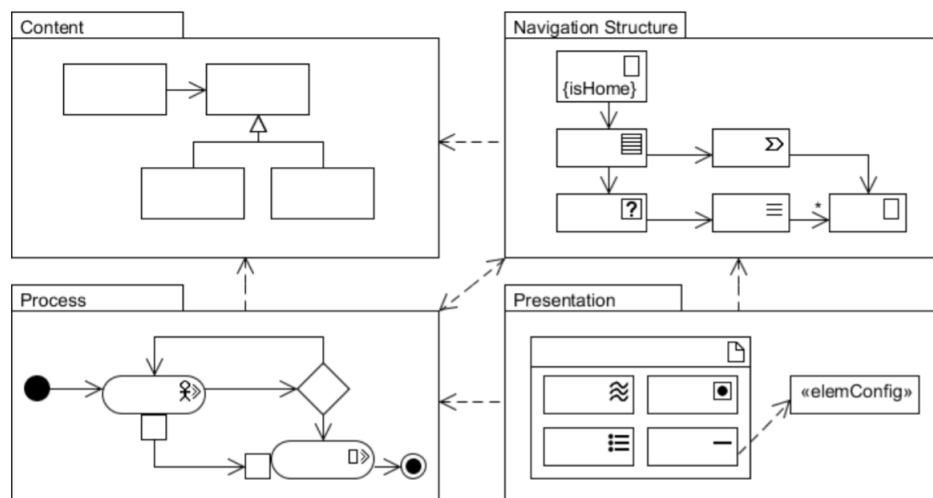


Figura 13. Vista General de Modelos UWE

Fuente: Universidad Gastón Dachary MagicUWE Transformations

2.16.4 Aplicaciones

La clasificación se puede realizar bajo ciertos criterios, como la complejidad de la aplicación, la complejidad de datos, estructura de datos, volatilidad, o la finalidad de la aplicación. La siguiente clasificación se basa en la finalidad de la aplicación:

- **Interactivas:** interacción con el usuarios netamente.
- **Transaccionales:** banca electrónica, compra electrónica.
- **Servicio:** orientadas al servicio, ej. Asistente financiera, simuladores.
- **Portales:** intermediarios en línea, centros comerciales de compra electrónica.
- **Informacionales:** difusión de información, independiente de personalización.
- **Descarga datos:** orientadas a la disponibilidad de material en servidores.
- **Comunidades:** servicio de subastas, foros de debate
- **Análisis de datos:** aplicaciones OLAP (por sus siglas en ingles de Online Analítica Processing).
- **Descarga de datos:** Servidores de material didáctico, servidores de canciones (Cantarero, 2016)

2.16.5 Fase de análisis de requisitos

2.16.5.1 Diagramas de casos de uso

La Fase de Análisis de Requerimientos realiza la captura de los mismos mediante diagramas de casos de uso acompañado de documentación que detallada.

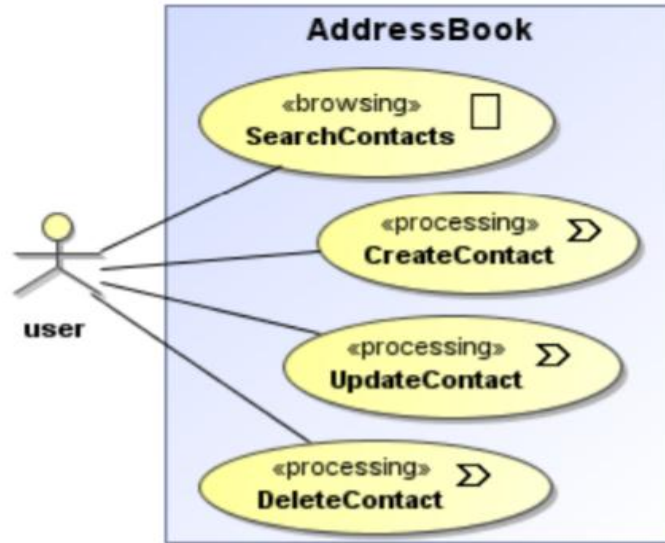


Figura 14. Diagrama de Caso de Uso

Fuente: Mamani, 2014

2.16.5.2 Fase Conceptual

Caracterizado por un modelo de dominio, que utiliza los requisitos que se detallan en los casos de uso. En esta etapa se representa el dominio del problema con un diagrama de clases de UML, que permiten determinar, métodos y atributos.

El propósito de este diagrama es construir un modelo del dominio que intenta no considerar el paseo de la navegación, la presentación y los aspectos de interacción. Aspectos que se analizarán en los pasos respectivos de navegación y presentación de la planificación.

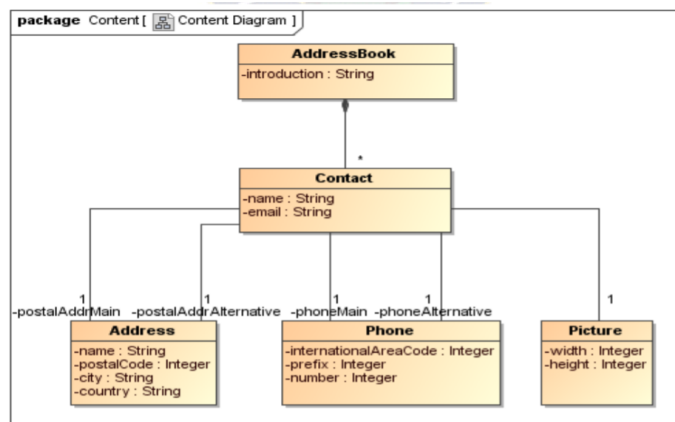


Figura 15. Diagrama de Modelo Conceptual

Fuente: Mamani, 2014

2.16.5.3 Fase De Navegación

Basado en el diagrama de la fase conceptual, donde se especifica los objetos que serán visitados dentro de la aplicación web y la relación entre los mismos.

Su objetivo principal es representar el diseño y estructura de las rutas de navegación al usuario para evitar la desorientación en el proceso de navegación.

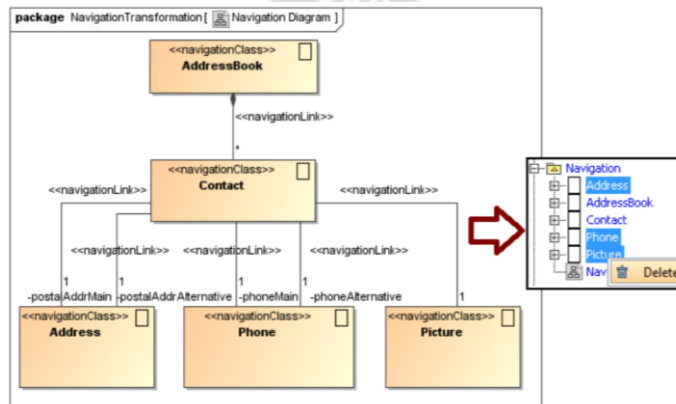


Figura 16. Diagrama de Modelo Navegacional

Fuente: Mamani, 2014

Este modelo se destaca en el marco de UWE como el más importante, ya que representa elementos estáticos, a la vez que se pueden incorporar lineamiento semántico de referencia para las funcionalidades dinámicas de una aplicación Web.

La fase de navegación a su vez podemos dividirlo en dos áreas:

- **Modelo del espacio de navegación:** basada en lo estructurado en la fase de conceptualización, es decir en los diagramas de clases.
- **Modelo de la estructura de navegación:** Muestra la forma de navegar ante el espacio de navegación. Están constituidas por menús, índices, visitas guiadas, y formularios.
 - Los índices es la colección de objetos permitiendo una navegación directa.
 - Las visitas guiadas compuesta por grupo de referencias, permitiendo una navegación secuencial.
 - Un menú es un elemento parte de la navegación con un número específico de conexiones a otros objetos.
 - Un formulario facilita al usuario ingresar información para completar las condiciones de selección de objetos pertenecientes a las colecciones de índices y visitas guiadas.



Figura 17. Estereotipos del Modelo Navegacional
Fuente: Mamani, 2014

a) Modelo De Estructura De Navegación

Este refinamiento consiste en mejorar el modelo d espacio de navegación añadiendo estructuras de acceso como: índices, guías de ruta, consultas y menús.

Las clases estereotipadas para estas estructuras son: <<index>> (especifica acceso directo a todas las instancias del destino proporcionando una lista de todos los elementos desde donde se puede seleccionar para la navegación en la aplicación web), <<guided tour>> (proporciona acceso secuencial a instancias del nodo destino), <<query>> (representa la posibilidad de buscar por instancias en el nodo destino), <<menu>> (son utilizados para estructurar la salida de enlaces desde un nodo).

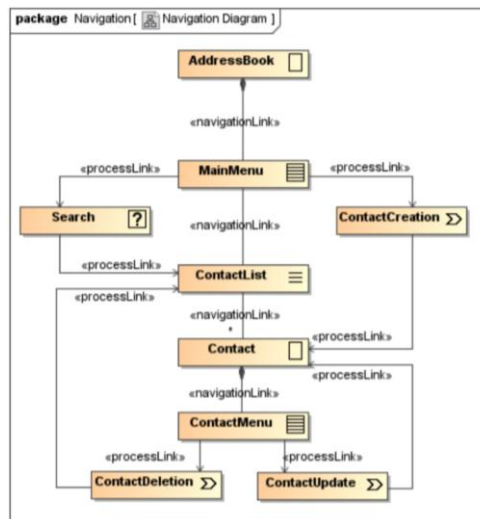


Figura 18. Diagrama de Modelo Estructura De Navegación
Fuente: Mamani, 2014

2.16.5.4 Fase De Diseño De Presentación

La fase de diseño de presentación tiene como objetivo la representación de las vistas del interfaz del usuario final, la representación gráfica de esta fase se encuentra basada en los diagramas realizados en las fases anteriores.

Las clases del modelo de presentación representan páginas Web o parte de ellas, organizando la composición de los elementos de la interfaz de usuario y las jerarquías del modelo de presentación.



Figura 19. Estereotipos del Modelo de Presentación.
Fuente: Mamani, 2014

El diagrama de esta fase representa los objetos de navegación y elementos de acceso, por ejemplo en que marco o ventana se encuentra el contenido y que será remplazado cuando se accione un enlace. En la siguiente imagen podremos observar un ejemplo de un diagrama de presentación mediante UWE.

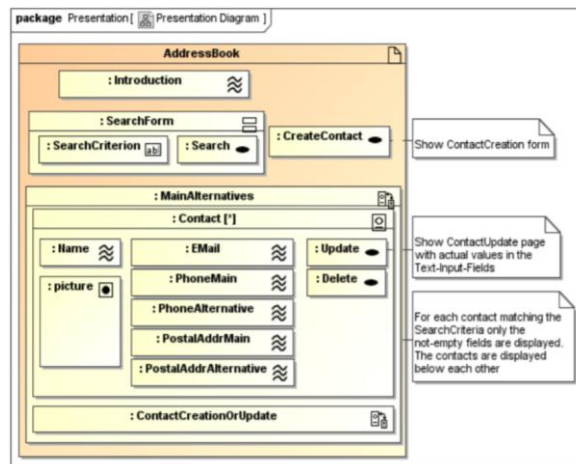


Figura 20. Diagrama de Modelo de Presentación
Fuente: Mamani, 2014

2.16.5.5 Modelo De Proceso

Le modelo de proceso o tareas integra los proceso de negocios al modelo de UWE, especificando los comportamientos de cada proceso y delas interfaces que permiten manejar a cada uno de ellos. Representa la parte dinámica de la aplicación Web, especificando la funcionalidad de las transiciones y de los flujos de trabajo complejos de las actividades, contrario al modelo navegación, que representa la parte estática de la información. (Mamani, 2014)

2.16.6 Diagramas

El diagrama de secuencias es un esquema conceptual que permite representar el comportamiento de un sistema, para lo cual emplea la especificación de los objetos que se encuentran en un escenario y la secuencia de mensajes intercambiados entre ellos, con el fin de llevar a cabo una transacción del sistema. Existen diferentes enfoques que buscan la generación automática de modelos conceptuales, como el diagrama de secuencias.

2.16.6.1 Elementos del diagrama de secuencias

El diagrama de secuencias hace parte de los diagramas de interacción de la especificación UML que describen los aspectos dinámicos de un sistema y muestran la interacción entre los objetos de un sistema y los mensajes enviados entre ellos, ordenados según su secuencia en el tiempo, sus elementos son:






Tipo de Nodo	Notación	Descripción
Marco		Provee un borde visual para el diagrama de secuencias
Línea de Vida		Representa un participante individual en una interacción
Actor		Representa el papel desempeñado por un usuario
Mensaje		Define una comunicación particular entre líneas de vida de una interacción
Fragmento combinado		Describe una interacción reutilizable

Figura 21. Elementos del Diagrama de Secuencias

Fuente: Jorge Paul Quispe Pila, 2013

Los diagramas de secuencias son útiles para diversos usos como:

- **El modelado de escenarios de uso.** Un escenario de uso es una descripción de una posible forma en que un sistema se utiliza. La lógica de un escenario de uso puede ser parte de un caso de uso, por ejemplo, una secuencia alternativa o un paso completo a través de un caso de uso, tal como la lógica que describe la secuencia normal de la acción o una parte de ella. Un escenario de uso también puede ser un paso a través de la lógica contenida en varios casos de uso.

- **El modelado de la lógica de los métodos.** Los diagramas de secuencias se pueden utilizar para explorar la lógica de una operación, función o procedimiento complejos, ya que ofrece una forma de observar las invocaciones a las operaciones definidas en las clases.
- **La detección de cuellos de botella** en un diseño orientado a objetos. Al observar los mensajes enviados a un objeto y cuánto se tardan en ejecutar el método invocado, es posible concluir que es necesario cambiar el diseño con el fin de distribuirla carga dentro del sistema.

2.16.7 Diagrama De Estado

Los diagramas de estado describen gráficamente los eventos y los estados de los objetos. Los diagramas de estado son útiles, entre otras cosas, para indicar los eventos del sistema en los casos de uso.

Un evento es un acontecimiento importante a tomar en cuenta para el sistema. Un estado es la condición de un objeto en un momento determinado: el tiempo que transcurre entre eventos. Una transición es una relación entre dos estados, e indica que, cuando ocurre un evento, el objeto pasa del estado anterior al siguiente.

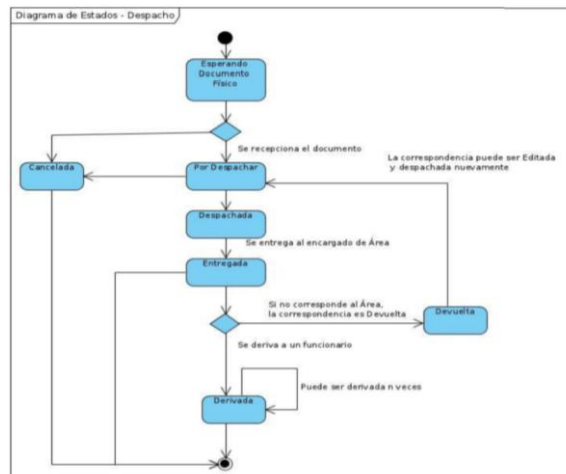


Figura 22. Diagrama de Estado

Fuente: Jorge Paul Quispe Pila, 2013

En UML, los estados se representan mediante óvalos. Las transiciones se representan mediante flechas con el nombre del evento respectivo. Se acostumbra poner un estado inicial y un final. En particular, es útil hacer diagramas de estado para describir la secuencia permitida de eventos en los casos de uso. (Pila, 2013)

2.17 HERRAMIENTAS

2.17.1 Php

En el texto Fundamentos de PHP refiere que, tres letras que juntas (PHP) constituyen el nombre de uno de los lenguajes de programación más populares para el desarrollo de Web.

Durante los últimos años, PHP se ha convertido, en la opción para el desarrollo de aplicaciones Web orientadas a bases de datos, sobre todo por su escalabilidad, facilidad, uso y el amplio soporte para diferentes bases de datos y formatos de éstos.

- **Características únicas**

Estás familiarizado con otros lenguajes que se ejecutan del lado del servidor, como ASP.NET o JSP, tal vez te preguntes qué tiene de especial PHP o qué lo hace tan diferente de esas opciones competidoras. Bien, he aquí algunas razones:

- **Portabilidad**

PHP está disponible para UNIX, Microsoft Windows, Mac OS y OS/2 y los programas escritos en PHP se pueden transportar de una plataforma a otra. Como resultado, las aplicaciones PHP desarrolladas en Windows, por ejemplo, se ejecutarán en UNIX sin grandes contratiempos. Esta capacidad de desarrollar fácilmente para múltiples plataformas es muy valiosa, en especial cuando se trabaja en un ambiente corporativo de varias plataformas o cuando se intenta atacar diversos sectores del mercado.

- **Fácil de usar**

También está la “La sencillez es la mayor sofisticación”, dijo Leonardo da Vinci y, de acuerdo con ello, PHP es un lenguaje de programación extremadamente sofisticado. Su sintaxis es clara y consistente y viene con una documentación exhaustiva para las más de 5 000 funciones incluidas en la distribución principal. Esto reduce de manera importante la curva de aprendizaje tanto para los desarrolladores novatos como para los expertos, y es una de las razones por las que PHP es favorecido como una herramienta rápida para la creación de prototipos que permitan el desarrollo de aplicaciones basadas en Web.

- **Código libre**

PHP es un proyecto de código libre; el lenguaje es desarrollado por un grupo de programadores voluntarios distribuidos por todo el mundo, quienes ponen a disposición gratuita el código fuente a través de Internet, y puede ser utilizado sin costo, sin pagos por

licencia y sin necesidad de grandes inversiones en equipo de cómputo ni programas. Con ello se reduce el costo del desarrollo de programas sin afectar la flexibilidad ni la confiabilidad de los productos. La naturaleza del código libre implica que cualquier desarrollador, dondequiera que se encuentre, puede inspeccionar el árbol de código, detectar errores y sugerir posibles correcciones; con esto se produce un producto estable y robusto, en que las fallas, una vez descubiertas, se corrigen rápidamente, en algunas ocasiones, ¡horas después de ser descubiertas!

- **Soporte comunitario**

Una de las mejores características de los lenguajes a los que da soporte una comunidad, como PHP, es el acceso que ofrece a la creatividad e imaginación de cientos de desarrolladores ubicados en diferentes partes del mundo. En la comunidad PHP, los frutos de esta creatividad pueden ser encontrados en PEAR (PHP Extension and Application Repository), el repositorio de extensiones y aplicaciones de PHP (<http://pear.php.net>), y en PECL (PHP Extension Community Library), la biblioteca de la comunidad de extensiones PHP (<http://pecl.php.net>), que contienen cientos de soluciones y extensiones que los desarrolladores pueden ocupar para añadir sin esfuerzo nuevas funcionalidades a sus aplicaciones PHP. Utilizar estas soluciones suele ser una mejor opción en tiempo y costo, en vez de desarrollar desde cero tu propio código.

- **Soporte a aplicaciones de terceros**

Una de las fortalezas históricas de PHP ha sido su soporte a una amplia gama de diferentes bases de datos, entre las cuales se incluyen MySQL, PostgreSQL, Oracle y Microsoft SQL Server. (Vaswani, 2010)

2.17.2 Base de datos

MariaDB es un sistema de gestión de base de datos con licencia GPL, derivado como fork o bifurcación de MySQL, líder tradicional en su segmento. En la actualidad hay más de seis millones de copias de MySQL funcionando, lo que supera la base instalada de cualquier otra herramienta de bases de datos. MySQL es una base de datos enormemente popular en aplicaciones web: suele estar vinculada a blogs como WordPress, Drupal, o en general, al uso de PHP. Desarrollada inicialmente por MySQL AB, compañía fundada por David Axmark, Allan Larsson y Ulf Michael «Monty» Widenius, MySQL carecía de algunos de los elementos considerados fundamentales en las bases de datos relacionales, como integridad referencial o

transacciones, pero a pesar de ello, atrajo a muchos desarrolladores de páginas web con contenido dinámico por su simplicidad. La evolución de su desarrollo ha ido cubriendo progresivamente estas carencias, y la llevaron a convertirse en el claro líder en su segmento. (Dans, 2013)

2.17.3 Servidor Web Apache

Apache es un software de servidor web gratuito y de código abierto con el cual se ejecutan el 46% de los sitios web de todo el mundo. El nombre oficial es Apache HTTP Server, y es mantenido y desarrollado por Apache Software Foundation.

Permitiendo a los propietarios de sitios web servir contenido en la web, de ahí el nombre de “servidor web”. Es uno de los servidores web más antiguos y confiables, con la primera versión lanzada hace más de 20 años, en 1995.

Cuando alguien quiere visitar un sitio web, ingresa un nombre de dominio en la barra de direcciones de su navegador. Luego, el servidor web envía los archivos solicitados actuando como un repartidor virtual.

Aquí en Hostinger, nuestra infraestructura de hosting web utiliza Apache en paralelo con NGINX, que es otro software popular de servidor web. Esta configuración particular nos permite obtener lo mejor de ambos mundos. Esto mejora en gran medida el rendimiento del servidor al compensar los lados más débiles de un software con las fortalezas del otro.

2.17.4 Servidor Web

Un servidor web es un programa de tipo informático que se encarga de procesar una aplicación del lado del servidor, cada una de las cuales puede acceder a archivos almacenados en un servidor físico y usarlos para diferentes propósitos, mediante conexiones bidireccionales o unidireccionales con la máquina del cliente, tras lo cual se genera una respuesta del lado del cliente. El trabajo de un servidor web es servir sitios web en Internet. Para lograr ese objetivo, actúa como un intermediario entre el servidor y las máquinas de los clientes. Extrae el contenido del servidor en cada solicitud de usuario y lo envía a la web.

El mayor desafío de un servidor web es servir a muchos usuarios diferentes de la web al mismo tiempo, cada uno de los cuales solicita diferentes páginas. Los servidores web procesan archivos escritos en diferentes lenguajes de programación como PHP, Python, Java y otros.

Los convierten en archivos HTML estáticos y le entregan estos archivos al navegador de los usuarios de la web. Cuando escuches la palabra servidor web, piensa que es la herramienta responsable de la comunicación adecuada entre el servidor y el cliente. (Gustavo, 2019)

2.17.5 Framework

Un Framework, es el esquema o estructura que se establece y que se aprovecha para desarrollar y organizar un software determinado. Esta definición, algo compleja, podría resumirse como el entorno pensado para hacer más sencilla la programación de cualquier aplicación o herramienta actual.

Este sistema plantea varias ventajas para los programadores, ya que automatiza muchos procesos y además facilita el conjunto de la programación. Es útil, por ejemplo, para evitar el tener que repetir código para realizar funciones habituales en un rango de herramientas, como puede ser el acceder a bases de datos o realizar llamadas a Internet. Todas estas tareas son las que se realizan de forma mucho más fácil cuando se trabaja dentro de un framework.

Plantea muchas ventajas y, además, es capaz de hacer que se realicen incluso labores mucho más complejas que, por otros medios, serían imposibles de plantear siquiera a la hora de programar algo. No obstante, su utilidad es algo que depende del tipo de programa y de contexto en el que vaya a emplearse.

Existen muchos frameworks diferentes, como pueden ser Meteor para trabajar con JavaScript en web y móviles o el más que popular .NET Framework de Microsoft. Todos ellos poseen herramientas específicas y funciones características que encajan mejor con el tipo de aplicación a desarrollar, así como el uso que se le pretenda dar una vez terminada. Son, a día de hoy, algo imprescindible para cualquier programador, sea amateur o profesional.

Para qué sirve un Framework

Un Framework sirve para poder escribir código o desarrollar una aplicación de manera más sencilla. Es algo que permite una mejor organización y control de todo el código elaborado, así como una posible reutilización en el futuro. Debido a esto, garantiza una mayor productividad que los métodos más convencionales y una minimización del coste al agilizar las horas de trabajo volcadas en el desarrollo.

Por otra parte, su acción es algo que afecta también a los errores, minimizándolos considerablemente. En definitiva, es algo que brinda una ayuda general y más que considerable al programador y desarrollador, haciendo que sus labores sean mucho más sencillas. (Isau, 2015)



Figura 23. Tipos de Framework

Fuente: Isau A. 2015

2.17.6 Laravel

Laravel, el “framework PHP para artesanos” según dicta su slogan. Laravel es un framework PHP de código abierto que intenta aprovechar las ventajas de otros Frameworks y desarrollar con las últimas versiones de PHP (entre otras muchas cosas que aporta como framework). Siendo un código PHP de forma elegante y simple basado en un modelo MVC (Modelo-Vista-Controlador).

Este framework está en constante mantenimiento y expansión por parte de sus desarrolladores lo que asegura la continuidad y seguridad del framework con actualizaciones regulares.

Con Laravel tienes todo lo necesario para empezar a usar seguridad OpenSSL y cifrado AES-256-CBC. Adicionalmente, todos los valores encriptados están firmados por un código de autenticación de mensaje “Token” que detecta si el mensaje encriptado fue alterado.

- **Características**

En esta sección, nos centraremos más en el apartado técnico de Laravel y en las ventajas que nos aporta a la hora de desarrollar aplicaciones web.

Las características más notables que aporta Laravel son las siguientes:

- **Blade**

Blade es un sistema de plantillas para crear vistas en Laravel. Este permite extender plantillas creadas y secciones en otras vistas en las cuales también tendremos accesibles las variables y con posibilidad de utilizar código PHP en ellas, además, ligado al uso de bootstrap u otro framework HTML generará resultados optimizados a los diferentes dispositivos (Móviles, Tablets, PC’s, etc..).

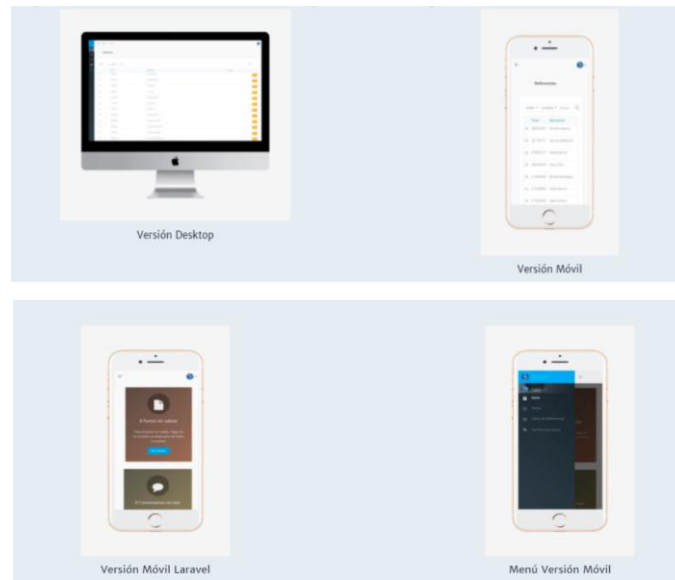


Figura 24. Características de Laravel

Fuente: Karim Sierra, 2018

- **Eloquent:** Eloquent es el ORM que incluye Laravel para manejar de una forma fácil y sencilla los procesos correspondientes al manejo de bases de datos en nuestro proyecto. Transforma las consultas SQL a un sistema MVC lo que no permite procesar consultas SQL directamente y así protegernos de la inyección SQL.
- **Routing:** Laravel proporciona un sistema de organización y gestión de rutas que nos permite controlar de manera exhaustiva las rutas de nuestro sistema.
- **Middlewares:** Son una especie de controladores que se ejecutan antes y después de una petición al servidor, lo que nos permite insertar múltiples controles, validaciones o procesos en estos puntos del flujo de la aplicación.
- **Comunidad y documentación:** Un gran punto a destacar de este framework es la gran comunidad y documentación que existe, una comunidad de profesionales activa que aporta conocimiento y funcionalidades, además de testear nuevas versiones y detectar fallos del framework, lo que le da seguridad al framework. Y una documentación muy completa y de calidad pensada para los propios desarrolladores.

Las posibles aplicaciones son todas las aplicaciones desarrolladas en PHP, por ejemplo, áreas cliente, intranets, aplicaciones web con funcionalidades concretas, APIS, y prácticamente cualquier funcionalidad web requiera programación a medida.

La potencia de Laravel radica en su integración, escalabilidad y facilidad de mantenimiento respecto a otros desarrollos en lenguajes 100% nativos y por lo tanto es una opción más que a tener en cuenta a la hora de decidir usar este framework en nuestros desarrollos.

- **Tres puntos a destacar de Laravel**

Los CMS no cubren nuestras necesidades como aplicación web.

Necesitamos de un desarrollo a medida ágil, seguro y con una fácil escalabilidad y mantenimiento.

Si además de necesitar una solución web, esta tiene que estar integrada con otras aplicaciones (Google, servicios de mailing, ERP's, etc.).

Llegados a este punto, y tras todo lo comentado, solo nos queda reafirmar que Laravel y el desarrollo con Frameworks está cogiendo cada vez más fuerza ya que ofrecen soluciones asequibles para pequeñas empresas sin necesidad de embarcarse en grandes proyectos faraónicos. (Sierra, 2018)

2.17.7 Javascript

Javascript es un lenguaje que puede ser utilizado por profesionales y para quienes se inician en el desarrollo y diseño de sitios web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos.

Muchos confunden el Javascript con el Java pero ambos lenguajes son diferentes y tienen sus características singulares. Javascript tiene la ventaja de ser incorporado en cualquier página web, puede ser ejecutado sin la necesidad de instalar otro programa para ser visualizado.

Java por su parte tiene como principal característica ser un lenguaje independiente de la plataforma. Se puede crear todo tipo de programa que puede ser ejecutado en cualquier ordenador del mercado: Linux, Windows, Apple, etc. Debido a sus características también es muy utilizado para internet.

Como síntesis se puede decir que Javascript es un lenguaje interpretado, basado en prototipos, mientras que Java es un lenguaje más orientado a objetos.

Un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript podemos crear diferentes efectos e interactuar con nuestros usuarios. Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows

y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros.

Es necesario resaltar que hay dos tipos de JavaScript: por un lado está el que se ejecuta en el cliente, este es el Javascript propiamente dicho, aunque técnicamente se denomina Navigator JavaScript. Pero también existe un Javascript que se ejecuta en el servidor, es más reciente y se denomina LiveWire Javascript. (Valdes, 2007)

2.18 PRUEBAS DE FUNCIONAMIENTO

2.18.1 Prueba de caja blanca

La prueba de caja blanca se basa en el diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivarlos. Mediante la prueba de la caja blanca el ingeniero del software puede obtener casos de prueba que:

- Garanticen que se ejerciten por lo menos una vez todos los caminos independientes de cada módulo, programa o método.
- Ejerciten todas las decisiones lógicas en las vertientes verdadera y falsa.
- Ejecuten todos los bucles en sus límites operacionales.
- Ejerciten las estructuras internas de datos para asegurar su validez.

Es por ello que se considera a la prueba de Caja Blanca como uno de los tipos de pruebas más importantes que se le aplican a los software, logrando como resultado que disminuya en un gran porcentaje el número de errores existentes en los sistemas y por ende una mayor calidad y confiabilidad.

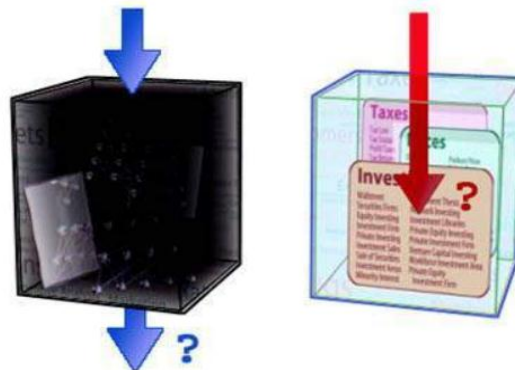


Figura 25. Prueba de caja blanca

Fuente: Pressman, Roger S

2.18.2 Prueba de caja negra

Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos.

La prueba de Caja Negra no es una alternativa a las técnicas de prueba de la Caja Blanca, sino un enfoque complementario que intenta descubrir diferentes tipos de errores a los encontrados en los métodos de la Caja Blanca. Muchos autores consideran que estas pruebas permiten encontrar:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a las Bases de Datos externas.
- Errores de rendimiento.
- Errores de inicialización y terminación.

Para desarrollar la prueba de caja negra existen varias técnicas, entre ellas están:

- Técnica de la Partición de Equivalencia: esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- Técnica del Análisis de Valores Límites: esta Técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- Técnica de Grafos de Causa-Efecto: es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

Dentro del método de Caja Negra la técnica de la Partición de Equivalencia es una de las más efectivas pues permite examinar los valores válidos e inválidos de las entradas existentes en el software, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de pruebas que descubran clases de errores, reduciendo así en número de clases de prueba que hay que desarrollar.

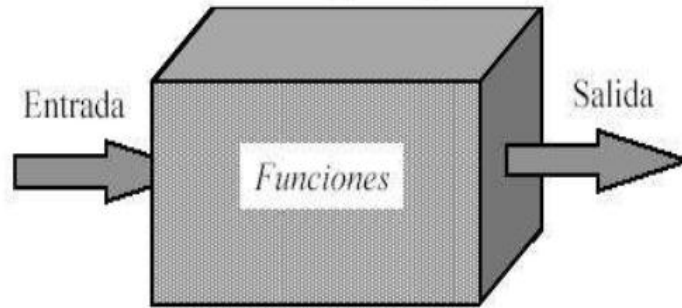


Figura 26. Prueba de caja negra
Fuente: Pressman, Roger S

2.19 METRICAS DE CALIDAD

El Modelo de Calidad de Software para la DGSC en el <http://www.dgsc.go.cr/>, refiere que, La norma ISO/IEC 9126 es un modelo de calidad estándar para productos de software. La primera parte del modelo especifica 6 características de **calidad interna y externa**, las cuales están divididas en sub características, son manifestadas externamente cuando el software es utilizado como parte de un sistema, y son un resultado de atributos internos del software.

- **La calidad externa**

Evalúa que el software satisfaga las necesidades del usuario teniendo en cuenta las condiciones especificadas. Esta calidad es medible en el comportamiento del producto.

- **La calidad interna**

Evalúa el total de atributos que un software debe satisfacer teniendo en cuenta condiciones especificadas.

Calidad de uso del producto



Figura 27. Proceso de Métricas de calidad
Fuente: Modelo De Calidad De Software para la DGSC

2.19.1 Características

Las características definidas son aplicables a todo tipo de software. Las características y sub características proveen una terminología consistente respecto de la calidad del producto del software. Esta Norma permite especificar y evaluar la calidad del software desde distintas perspectivas, las cuales están asociadas a la adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad.

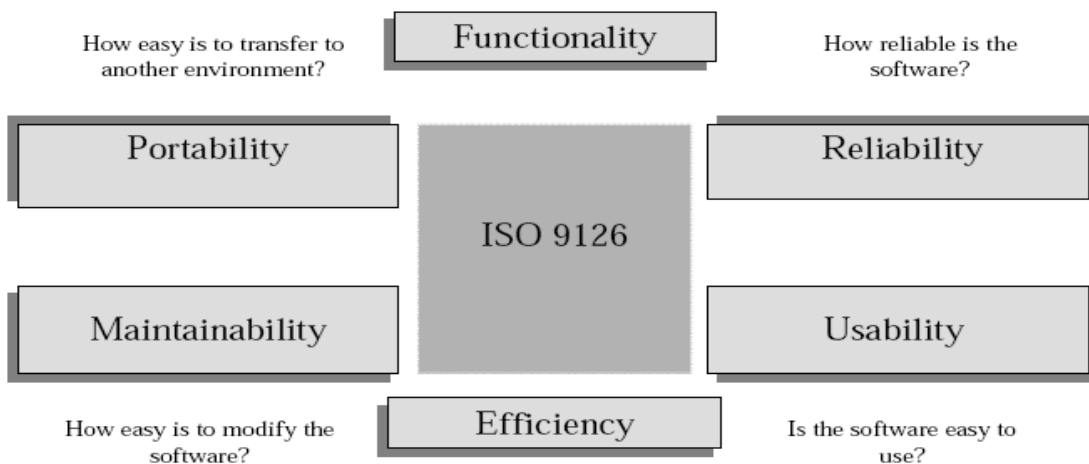


Figura 28. Características de ISO

Fuente: Modelo de Calidad de Software para la DGSC

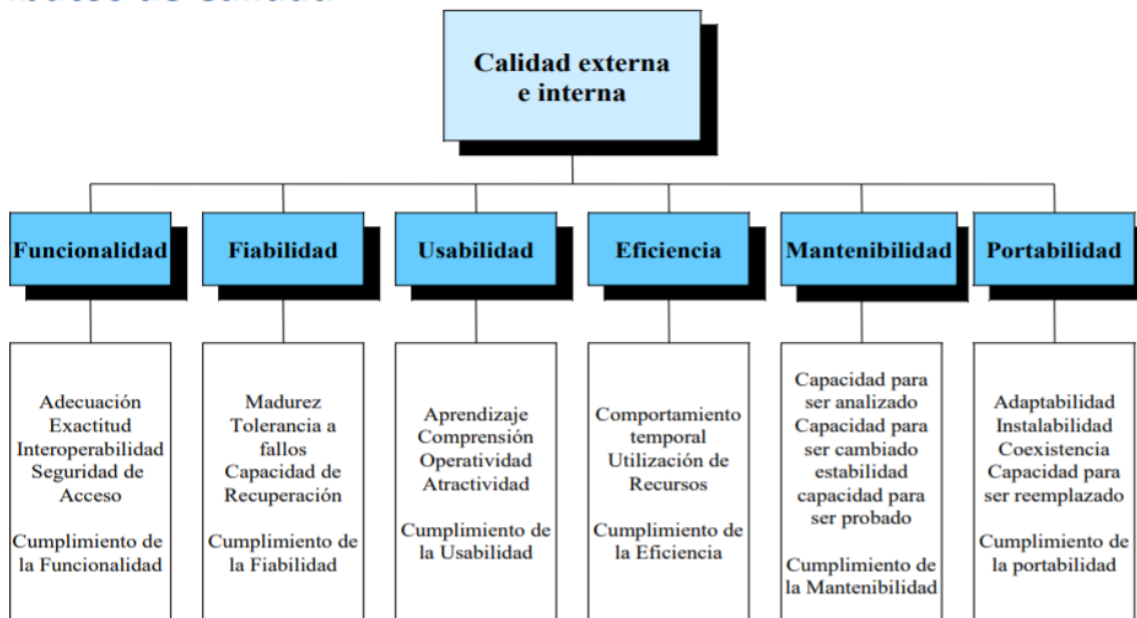


Figura 29. Sub Características de la Norma ISO/IEC 9126

Fuente: Atributos de calidad

2.19.1.1 Funcionalidad

Un conjunto de atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen las necesidades implícitas o explícitas. Sus atributos son: adecuación, exactitud, interoperabilidad, seguridad y cumplimiento funcional.

Tabla 2.
Dominios de información de puntos de función

DOMINIO DE INFORMACIÓN	DESCRIPCIÓN
Número de entradas del usuario	Se encuentra cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas se deberían diferenciar de las peticiones, las cuales se cuentan de forma separada.
Número de salidas del usuario	Se cuenta cada salida que proporciona al usuario información orientada a la aplicación, en este contexto la salida se refiere a informes, pantallas, mensajes de error y demás. Los elementos de datos particulares dentro de un informe no se encuentran de forma separada.
Número de peticiones al usuario	Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.
Número de archivos	Se cuenta cada archivo maestro lógico (esto es un grupo lógico de datos que se puede ser una parte de una gran base de datos o un archivo independiente).
Número de interfaces externas	Se cuenta todas las interfaces legibles por la maquina (por ejemplo archivos de datos de disco), que se utilizan para transmitir información a otros sistemas.

Fuente: *Elaboración propia*

Los puntos de función se calculan completando la siguiente tabla:

Tabla 3.
Factores de ponderación

PARÁMETROS DE MEDICIÓN	CUENTA	SIMPLE	MEDIO	COMPLEJO	RESULTADO
Número de entradas de usuario	N_1	3	4	6	$N_1 * \text{factor}$
Número de salidas de usuario	N_2	4	5	7	$N_2 * \text{factor}$
Número de peticiones de usuario	N_3	3	4	6	$N_3 * \text{factor}$

Número de archivos	N ₄	7	10	15	N ₄ *factor
Número de interfaces externas	N ₅	5	7	10	N ₅ *factor
Cuenta total					Σ(N_i*factor)

Fuente: *Elaboración propia*

Para calcular puntos de función (PF) se utiliza la siguiente relación:

$$PF = \text{cuenta total} * [X + (\text{Min} * \sum (F_i))]$$

En donde cuenta total es la suma de todas las entradas de los factores de ponderación obtenidas en la tabla anterior.

F_i (i = 1 a 14), son valores de ajuste de complejidad según las respuestas a las siguientes preguntas:

Tabla 4.
Valores de ajuste de la complejidad

IMPORTANCIA	0%	20%	40%	60%	80%	100%
Escala	No influencia	Incidencial	Moderado	Medio	Significativo	Esencial
Factor	0	1	2	3	4	5

¿Requiere el sistema copias de seguridad y recuperación fiables?

¿Se requiere comunicación de datos?

¿Existen funciones de procesamiento distribuido?

¿Es crítico el rendimiento?

¿Se ejecuta el sistema en un entorno operativo existente y fuertemente utilizado?

¿Requiere el sistema entrada de datos interactiva?

¿Requiere el sistema entrada de datos interactivos que las transacciones de entrada se lleven a cabo sobre múltiples entradas u operaciones?

¿Se actualizan los archivos maestros de forma interactiva?

¿Son complejas las entradas, las salidas, los archivos o peticiones?

¿Es complejo el procesamiento interno?

¿Se ha diseñado código para ser reutilizable?

¿Están incluidas en el diseño la conversión y la instalación?

¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?

¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?

Fuente: *Elaboración propia*

También la métrica de adecuación según la siguiente tabla:

Tabla 5.
Métrica de adecuación

NOMBRE:	COMPLETITUD DE IMPLEMENTACIÓN FUNCIONAL
Propósito:	Cómo de completa es la implementación funcional. Contar las funciones faltantes detectadas en la evaluación y
Método de aplicación:	comparar con el número de funciones descritas en la especificación de requisitos.
	$X = 1 - A/B$
Medición, fórmula:	A = número de funciones faltantes B = número de funciones descritas en la especificación de requisitos
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, más completa.
Tipo de escala:	Absoluta $X = \text{count}/\text{count}$
Tipo de medida:	A = count B = count

	Especificación de requisitos
Fuente de medición:	Diseño
	Código fuente
	Informe de revisión

Fuente: *Elaboración propia*

2.19.1.2 Fiabilidad

Se refiere a un conjunto de atributos que miden la capacidad que tiene el software para mantener un nivel de rendimiento óptimo, bajo determinadas condiciones y durante un periodo de tiempo determinado. Sus atributos son madurez, tolerancia a fallos y la capacidad de recuperación ante un fallo.

Para que un sistema sea fiable, se debe garantizar un nivel de seguridad. La seguridad se subdivide a su vez en confidencialidad, autenticación, control de acceso, integridad de los datos y responsabilidad de los usuarios. Para garantizarla se ofrecen distintos mecanismos como certificados digitales y sockets (SSL) y hace un tratamiento adecuado de la información personal y privada de los usuarios.

La confiabilidad de un sistema se calcula mediante la siguiente relación

Dónde:

$F_c = 0,87$: funcionalidad del sistema

$\lambda = 1$: tasa de fallos dentro de un mes

$$F(t) = F_c * (e^{-\lambda * t})$$

Con:

Probabilidad de hallar una falla: $P(T \leq t) = F(t)$

Probabilidad de no hallar una falla: $P(T > t) = 1 - F(t)$

También utilizando la métrica de madurez de la siguiente tabla:

Tabla 6.
Métrica de madurez

NOMBRE:	SUFICIENCIA DE LAS PRUEBAS
Propósito:	Cuántos de los casos de prueba necesarios están cubiertos por el plan de pruebas.

Método de aplicación:	Contar las pruebas planeadas y comparar con el número de pruebas requeridas para obtener una cobertura adecuada. $X = A/B$
Medición, fórmula:	A = número de casos de prueba en el plan B = número de casos de prueba requeridos $0 \leq X$
Interpretación:	Entre X sea mayor, mejor la suficiencia.
Tipo de escala:	Absoluta $X = \text{count}/\text{count}$
Tipo de medida:	A = count B = count
Fuente de medición:	A proviene del plan de pruebas B proviene de la especificación de requisitos
ISO/IEC 12207 SLCP:	Aseguramiento de Calidad Resolución de problemas Verificación
Audiencia:	Desarrolladores Mantenedores

Fuente: *elaboración propia*

2.19.1.3 Usabilidad

Se refiere a un conjunto de atributos que miden el esfuerzo cognitivo necesario que deben realizar los usuarios para utilizar el sistema de software. Sus atributos son comprensión, curva de aprendizaje y operatividad.

Utilizando la métrica de entendibilidad según la siguiente tabla:

Tabla 7.
Métrica de entendibilidad

NOMBRE:	FUNCIONES EVIDENTES
Propósito:	Qué proporción de las funciones del sistema son evidentes al usuario.
Método de aplicación:	Contar las funciones evidentes al usuario y comparar con el número total de funciones.

	$X = A/B$
Medición, fórmula:	A = número de funciones (o tipos de funciones) evidentes al usuario
	B = total de funciones (o tipos de funciones)
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mejor.
Tipo de escala:	Absoluta
	$X = \text{count}/\text{count}$
Tipo de medida:	A = count
	B = count
	Especificación de requisitos
Fuente de medición:	Diseño
	Informe de revisión
ISO/IEC 12207	Verificación
SLCP:	Revisión conjunta

Fuente: Elaboración propia

2.19.1.4 Eficiencia

Se refiere a un conjunto de atributos que miden la relación entre el rendimiento del software y la cantidad de recursos utilizados, dada una situación determinada. Sus atributos son tiempo de respuesta y recursos utilizados

La eficiencia se entiende como la capacidad del sistema para proporcionar tiempos de respuesta, tiempos de proceso y potencia apropiados bajo condiciones determinadas.

Utilizando la métrica de comportamiento en el tiempo según la siguiente tabla:

Tabla 8.
Métrica de comportamiento en el tiempo

NOMBRE:	TIEMPO DE RESPUESTA
Propósito:	Cuál es el tiempo estimado para completar una tarea.
Método de aplicación:	Evaluar la eficiencia de las llamadas al SO y a la aplicación. Estimar el tiempo de respuesta basado en ello. Puede medirse: Todo o partes de las especificaciones de diseño.

	Probar la ruta completa de una transacción.
	Probar módulos o partes completas del producto.
	Producto completo durante la fase de pruebas.
Medición, fórmula:	X = tiempo (calculado o simulado)
Interpretación:	Entre más corto, mejor.
Tipo de escala:	Proporción
Tipo de medida:	X = time
Fuente de medición:	Sistema operativo conocido Tiempo estimado en llamadas al sistema
ISO/IEC 12207	Verificación
SLCP:	Revisión conjunta
Audiencia:	Desarrolladores Requeridores

Fuente: *Elaboración propia*

2.19.1.5 Mantenibilidad

Se refiere a un conjunto de atributos relacionados con el esfuerzo necesario para realizar determinadas modificaciones en el producto. Sus atributos son la capacidad de ser analizado, capacidad para ser modificado, estabilidad y capacidad para ser probado.

El estándar IEEE 982.1 sugiere un índice de madurez del software (IMS) que proporciona una indicación de la estabilidad del producto de software, se determina con la siguiente relación

$$IMS = [M_T - (F_c + F_a + F_d)] / M_T$$

A medida que el IMS se aproxima a 1.0 se logra una madurez estable.

Utilizando la métrica de cambiabilidad según la siguiente tabla:

Tabla 9.
Métrica de cambiabilidad

NOMBRE:	REGISTRO DE CAMBIOS
Propósito:	¿Se registran adecuadamente los cambios a la especificación y a los módulos con comentarios en el código?

Método de aplicación:	Registrar la proporción de información sobre cambios a los módulos $X = A/B$
Medición, fórmula:	A = número de cambios a funciones o módulos que tienen comentarios confirmados B = total de funciones o módulos modificados $0 \leq X \leq 1$
Interpretación:	Entre más cercano a 1, más registrable. 0 indica un control de cambios deficiente o pocos cambios y alta estabilidad.
Tipo de escala:	Absoluta $X = \text{count}/\text{count}$
Tipo de medida:	A = count B = count Sistema de control de configuraciones
Fuente de medición:	Bitácora de versiones Especificaciones
ISO/IEC 12207	Verificación
SLCP:	Revisión conjunta Desarrolladores
Audiencia:	Mantenedores Requeridores

Fuente: *Elaboración propia*

2.19.1.6 Portabilidad

Son atributos con la capacidad del software de ser transferido de un entorno a otro. Sus atributos son adaptabilidad, capacidad de instalación, coexistencia y capacidad de reemplazamiento.

Utilizando la métrica de conformidad de transportabilidad según la siguiente tabla:

Tabla 10.
Métrica de conformidad de transportabilidad

NOMBRE:	CONFORMIDAD DE TRANSPORTABILIDAD
----------------	---

Propósito:	Cómo de transportable es el producto según las regulaciones, estándares y convenciones aplicables.
Método de aplicación:	Contar los artículos encontrados con conformidad y comparar con el número de artículos en la especificación que requieren conformidad. $X = A/B$
Medición, fórmula:	A = número de artículos implementados de conformidad B = total de artículos que requieren conformidad $0 \leq X \leq 1$
Interpretación:	Entre más cercano a 1, más completa.
Tipo de escala:	Absoluta $X = \text{count}/\text{count}$
Tipo de medida:	A = count B = count Especificación de conformidad y estándares, convenciones y regulaciones relacionados.
Fuente de medición:	Diseño Código fuente Informe de revisión
ISO/IEC 12207	Verificación
SLCP:	Revisión conjunta
Audiencia:	Requeridores Desarrolladores

Fuente: *Elaboración propia*

2.19.1.7 Análisis comparativo de parámetros

La Tabla presenta una comparativa de las principales características de calidad de la norma ISO/IEC 9126, para una posterior selección de parámetros que ayuden en el análisis comparativo de la investigación.

NRO.	PARÁMETRO	INDICADOR	APLICA	NO APLICA
1	Funcionalidad	Adecuación	✓	
		Exactitud		x
		Operabilidad		x
		Seguridad de acceso		x
		Cumplimiento de Funcionalidad		x
2	Confiabilidad (Fiabilidad)	Madurez		x
		Tolerancia a fallos		x
		Recuperabilidad		x
		Cumplimiento de Fiabilidad		x
3	Factibilidad de uso (Usabilidad)	Capacidad de ser entendido	✓	
		Capacidad de ser aprendido	✓	
		Capacidad de ser operado	✓	
		Capacidad de atracción	✓	
		Cumplimiento de usabilidad	✓	
4	Eficiencia	Comportamiento temporal	✓	
		Utilización de recursos	✓	
		Cumplimiento de eficiencia	✓	
5	Mantenibilidad	Facilidad de análisis		x
		Estabilidad		x
		Facilidad de prueba		x
		Capacidad de ser cambiado		x
		Adaptabilidad		x
6	Portabilidad	Facilidad de instalación		x
		Coexistencia		x
		Reemplazabilidad	✓	

Figura 30. Tabla de parámetros

Fuente: Modelo de Calidad de Software para la DGSC

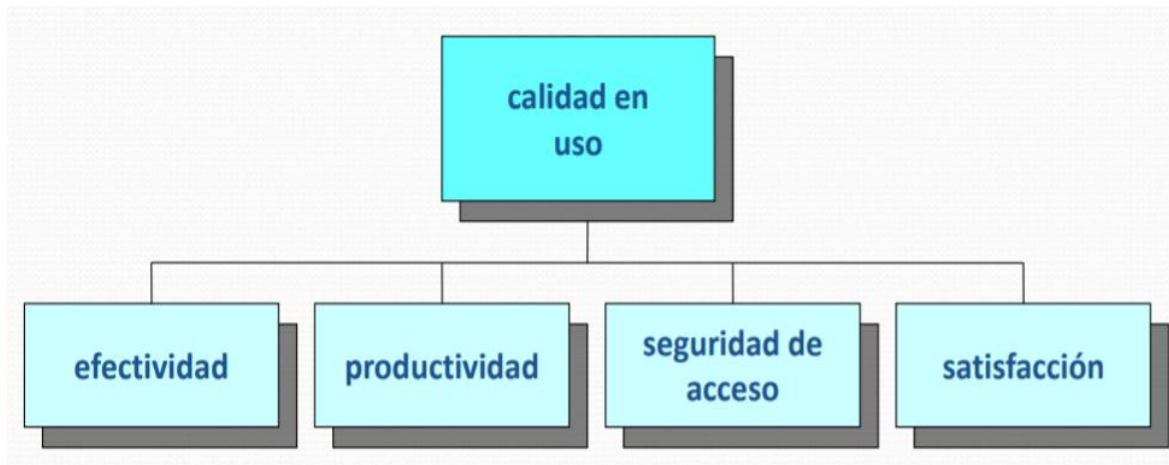


Figura 31. Modelo de calidad de uso

Fuente: Modelo de Calidad de Software para la DGSC

- **Efectividad**, Permitir a los usuarios alcanzar objetivos especificados con exactitud y completitud, en un contexto de uso especificado.
- **Productividad**, Permitir a los usuarios gastar una cantidad adecuada de recursos con relación a la efectividad alcanzada, en un contexto de uso especificado.

- **Seguridad física**, Alcanzar niveles aceptables del riesgo de hacer daño a personas, al negocio, al software, a las propiedades o al medio ambiente en un contexto de uso especificado.
- **Satisfacción**, Satisfacer a los usuarios en un contexto de uso especificado.

(Modelo de Calidad de Software para la DGSC, 2011)

2.19.2 Criterios de evaluación

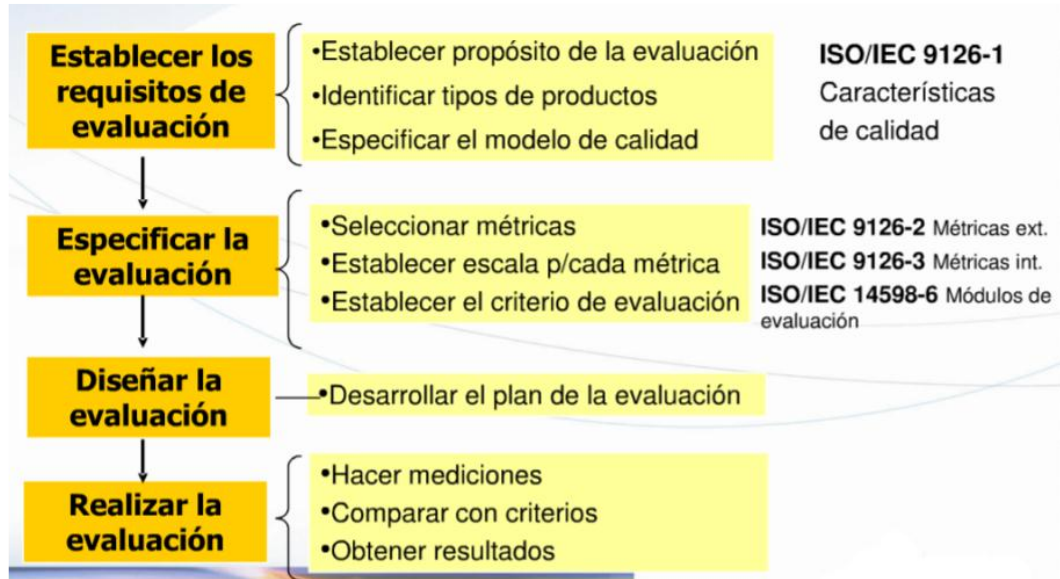


Figura 32. *Procesos de Evaluación*
Fuente: *Proceso de Evaluación*

2.20 COSTOS

2.20.1 Conceptos

En el desarrollo de este capítulo, se dará a conocer a la empresa LINXS S.R.L. que en la implementación y utilización del sistema se obtendrán muchos beneficios. Para tal efecto se realiza el análisis de costo del sistema usando el método COCOMO II y en base al costo del sistema y otros gastos se determina la rentabilidad del sistema con el cálculo del valor neto actual y la tasa interna de retorno. Después de realizar los cálculos necesarios para la obtención de los resultados esperados estaremos en la capacidad de afirmar si el proyecto es viable, redituable y comprobar que es buena opción invertir en el proyecto.

2.20.2 Cócomo II

El COCOMO II es un modelo empírico que se derivó al recopilar datos a partir de un gran número de proyectos de software. Dichos datos se analizaron para descubrir qué fórmulas se ajustaban mejor con las observaciones. Dichas fórmulas vinculan el tamaño del sistema y los factores del producto, proyecto y equipo, con el esfuerzo para desarrollar el sistema. COCOMO II es un modelo de estimación bien documentado y no registrado. COCOMO II se desarrolló a partir de los primeros modelos de estimación de costos COCOMO, que se basaron principalmente en el desarrollo de código original (Boehm, 1981; Boehm y Royce, 1989). El modelo COCOMO II toma en cuenta enfoques más modernos para el desarrollo de software, tales como el desarrollo rápido que usa lenguajes dinámicos, el desarrollo mediante composición de componentes y el uso de programación de base de datos. COCOMO II soporta el modelo en espiral de desarrollo, e incrusta sub modelos que producen estimaciones cada vez más detalladas.

Pertenece a la categoría de modelos de subestimaciones basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el "tamaño" del proyecto, en líneas de código principalmente.

COCOMO II tiene cuatro sub modelos: modelo de composición de aplicación, modelo de diseño temprano, modelo de reutilización y modelo post-arquitectónico.

2.20.3 Modelo Post-Arquitectónico

El modelo post arquitectura se utiliza en la fase de construcción del producto software, fase en la cual se tiene a disposición varios parámetros necesarios para aplicar este modelo de COCOMO II.

“El modelo pos arquitectónico es el más detallado de los modelos COCOMO II. Se usa una vez que está disponible un diseño arquitectónico inicial para el sistema, de manera que se conoce la estructura del subsistema. Entonces es posible hacer estimaciones para cada parte del sistema”.

“Como todos los modelos de estimación para software, los modelos COCOMO II requieren información sobre el dimensionamiento del software. Como parte de la jerarquía del modelo, están disponibles tres diferentes opciones de dimensionamiento: puntos objeto, puntos de función y líneas de código fuente”. (Roger P. R., 2010)

Por lo tanto, el modelo post arquitectura del modelo COCOMO II requiere información detallada de la estructura, del dimensionamiento del producto software.

Las ecuaciones que se utilizan son:

$$E = a(Kl)^b * m(X), \text{ en persona-mes.}$$

$$Tdev = c(E)^d, \text{ en meses.}$$

$$P = E/Tdev, \text{ en personas.}$$

Dónde:

E = es el esfuerzo requerido por el proyecto, en persona-mes.

Tdev = es el tiempo requerido por el proyecto, en meses.

P = es el número de personas requerido por el proyecto.

a, b, c y d = constantes con valores definidos en una tabla, según cada sub modelo.

Kl = es la cantidad de líneas de código, en miles.

m(X) = Es un multiplicador que depende de 15 atributos.

2.20.4 Constantes de complejidad

Se utiliza para obtener una primera aproximación rápida del esfuerzo, y hace uso de la siguiente tabla de constantes para calcular distintos aspectos de costes:

Tabla 11.
Constantes de complejidad

MODO	A	B	C	D
Orgánico	2.40	1.05	2.50	0.38
Semi-acolpado	3.00	1.12	2.50	0.35
Rígido	3.60	1.20	2.50	0.32

Fuente: *Pressman, 2005*

2.20.5 Factores multiplicadores de esfuerzo

Cada factor se cuantifica para un entorno de proyecto. La escala es;

- muy baja
- bajo
- nominal

- alto
- muy alto
- extremadamente alto.

Dependiendo de la calificación de cada atributo, se asigna un valor para usar de multiplicador en la fórmula (por ejemplo, si para un proyecto el atributo *DATA* es calificado como *muy alto*, el resultado de la fórmula debe ser multiplicado por 1000).

El significado de los atributos es el siguiente, según su tipo:

- **De software**

- **RELY**

Garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto. Va desde la sola inconveniencia de corregir un fallo (*muy bajo*) hasta la posible pérdida de vidas humanas (*extremadamente alto*, software de alta criticidad).

- **DATA**

Tamaño de la base de datos en relación con el tamaño del programa. El valor del modificador se define por la relación: D/K donde D corresponde al tamaño de la base de datos en bytes y K es el tamaño del programa en cantidad de líneas de código.

- **CPLX**

Representa la complejidad del producto.

- **De hardware**

- **TIME**: limitaciones en el porcentaje del uso de la CPU.

- **STOR**: limitaciones en el porcentaje del uso de la memoria.

- **VIRT**: volatilidad de la máquina virtual.

- **TURN**: tiempo de respuesta requerido.

- **De personal**

- **ACAP**: calificación de los analistas.

- **AEXP**: experiencia del personal en aplicaciones similares.

- **PCAP**: calificación de los programadores.

- **VEXP**: experiencia del personal en la máquina virtual.

- **LEXP**: experiencia en el lenguaje de programación a usar.

- **De proyecto**

- **MODP:** uso de prácticas modernas de programación.
- **TOOL:** uso de herramientas de desarrollo de software.
- **SCED:** limitaciones en el cumplimiento de la planificación.

El valor de cada atributo, de acuerdo a su calificación:

Tabla 12.
Variables factor de ajustes del esfuerzo

ATRIBUTOS	VALOR					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos de software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de hardware						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
Atributos del proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	

Fuente: *Elaboración propia*

2.21 SEGURIDAD DEL SISTEMA DE LA INFORMACION

Por seguridad de la información se entiende el conjunto de medidas preventivas y reactivas que permiten resguardar y proteger la información. Dicho de otro modo, son todas aquellas políticas de uso y medidas que afectan al tratamiento de los datos que se utilizan en una organización. La

seguridad de la información, como concepto, se basa en cuatro pilares: la disponibilidad, la integridad, la confidencialidad y la autenticación.

- **Disponibilidad:** Acceso a la información cuando se requiere, teniendo en cuenta la privacidad. Evitar “caídas” del sistema que permitan accesos ilegítimos, que impidan el acceso al correo.
- **Confidencialidad:** Información accesible solo para personal autorizado. La información no debe llegar a personas o entidades que no estén autorizados.
- **Integridad:** Información correcta sin modificaciones no autorizadas ni errores. Se protege frente a vulnerabilidades externas o posibles errores humanos.
- **Autenticación:** Información procedente de un usuario que es quien dice ser. Se verifica y se debe garantizar que el origen de los datos es correcto.

2.21.1 ISO/IEC-27000

La ISO 27000 es una norma que define de qué manera se debe implantar un Sistema de Gestión de la Seguridad de la Información en una empresa u organización. 48 Su implantación ofrece a la organización o empresa la ventaja de proteger su información de la forma más fiable posible, persiguiéndose para ello un total de tres objetivos principales:

- Preservar la confidencialidad de sus datos.
- Conservar la integridad de sus datos.
- Disponibilidad de la información protegida.

No en vano, tal es su importancia que la implantación de este tipo de herramienta garantiza que los riesgos de seguridad de la información son controlados por la organización eficientemente, tanto de forma interna como al resto de las empresas.

2.21.2 Encriptación AES-256

AES (Advanced Encryption Standard) significa Estándar de Cifrado Avanzado, es uno de los algoritmos de cifrado más utilizados y seguros actualmente disponibles la cual sigue siendo el único algoritmo en la lista del National Institute of Standards and Technology (NIST) para proteger datos clasificados.

AES es lo que se conoce como un cifrado simétrico por bloques, para ello, utiliza una clave criptográfica específica, que es efectivamente un conjunto de protocolos para manipular información. Esta clave puede ser de 128, 192 o 256 bits de tamaño.

AES-256 la versión clave de 256 bits de AES es el estándar de cifrado utilizado por Laravel. Es la forma más avanzada del cifrado y consiste en 14 rondas de sustitución, transposición y mezcla para un nivel de seguridad excepcionalmente alto.

2.21.3 Seguridad de la base de datos

Se usó como base de datos MariaDB. En cuanto a la forma de resguardo se realiza:

- Cuando una acción del usuario en el sistema requiere o solicita algunos registros de la base de datos, existe una conexión segura para esta acción.
- Para la seguridad de datos del sistema se tienen registrado de nombre de usuario y contraseña de acceso, según su nivel de acceso pueda realizar actividades en el sistema.

La información en una Institución es muy valiosa, por tanto su resguardo es fundamental, la conexión a la base de datos y el cierre de la conexión es de forma automática. En cuanto a las amenazas de SQL Inyección que es una de las más comunes amenazas, el framework Laravel incorpora un ORM para que el desarrollador deje de preocuparse por este tipo de ataques ya que el ORM está basado en una capa de objetos y así no ser capaz de interpretar el lenguaje SQL.

CAPÍTULO III
MARCO APLICATIVO

3.1 INTRODUCCION

En este capítulo se efectúa el diseño y desarrollo e implementación del sistema correspondiente. Para poder aplicar la ingeniería de requisitos, se aplica las actividades correspondientes a conceptualización, análisis, diseño y desarrollo del sistema de acuerdo a procesos.

Para el desarrollo del sistema se escoge la metodología ágil UWE, que utiliza un modelo de desarrollo incremental, y este se complementa con la metodología UML para las etapas de desarrollo.

El uso de la metodología UWE implica la realización de actividades por cada una de sus fases, a continuación, se describe por cada fase de la metodología, las actividades que se realizaran a lo largo de todo el desarrollo de este capítulo.

3.2 ANALISIS DE REQUERIMIENTO

El análisis de requerimientos funcionales y no funcionales pretende dar una guía de cómo se tomaron y trataron los requisitos de este proyecto. Además de servir como apoyo para usuarios, administradores o desarrolladores.

La captura de requerimientos es la actividad mediante la que el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema a desarrollar. El proceso de captura de requisitos puede resultar complejo, principalmente si el entorno de trabajo es desconocido para el equipo de analistas, y depende mucho de las personas que participen en él

La técnica que se usó para la captura de requisitos en el sistema de registro geolocalización de viveros en la ciudad de La Paz, mediante la colaboración del personal involucrado se consiguió establecer que se necesitaba un sistema que permita no solo digitalizar la información, sino también tener la capacidad de realizar un seguimiento de los viveros existentes en el departamento de la Paz.

3.2.1 Requerimientos funcionales

En los requerimientos funcionales actúan con información obtenida y su análisis, obtuvimos las funciones con las que el sistema deberá cumplir para satisfacer las necesidades de la institución y sus Clientes.

Tabla 13.
Representación de requerimientos funcionales

FUNCIÓN	CATEGORÍA
Acceder a la aplicación por tipos o niveles	Evidente
Registrar datos existentes	Evidente
Registrar usuarios al sistema	Evidente
Registro de viveros	Evidente
Registro de entidades financieras	Evidente
Generar listado de los viveros	Evidente
Generar listado de los usuarios	Evidente
Validación de entrada de datos	Oculto
Inicio y cierre de sesión	Oculto
Almacenamiento digital de información	Evidente
Geolocalización de viveros	Evidente
Detalle de los viveros	Evidente

Fuente: *Elaboración propia*

3.2.2 Requerimientos no funcionales

En los requerimientos no funcionales se muestran funciones pertenecientes al sistema que no actúan como comportamientos del mismo, sin embargo toman un valor significativo en su desarrollo, sumando características que apoyaran a que este se adopte más fácilmente por los usuarios. (Koch Nora, 2001)

Tabla 14.
Requerimientos no funcionales

FUNCIÓN	CATEGORÍA
El sistema deberá tener funcionalidad en cualquier navegador de internet como, Internet explore, Chrome, Firefox y otros	Evidente
Revisiones periódicas del sistema, para mantener el buen funcionamiento del mismo	Evidente
Respaldo de la información almacenada en el sistema	Evidente

Fuente: *Elaboración propia*

3.3 DISEÑO

En esta sección se presentan los casos de uso del sistema, los cuales describen la secuencia de eventos que realiza un actor cuando el sistema lleva a cabo un proceso.

Además, proporciona un medio por el cual las personas involucradas en el sistema, tanto los usuarios finales como el equipo de desarrollo, lleguen a una comprensión de éste.

3.3.1 Diagrama de caso de uso general

En la figura siguiente se muestra el diagrama del caso de uso general del sistema para el control de registro de geolocalización de viveros de la entidad.



Figura 33. Caso de uso General
Fuente: Elaboración Propia

3.3.2 Caso de uso: iniciar sesión

En la figura se muestra el inicio de sesión, describe como un usuario ingresa al sistema web, para previamente acceder a la página de ingreso del sistema, ingresa su nombre y contraseña, si los datos son correctos ingresa al sistema, si no muestra un mensaje de error.

Figura 3.1: Caso de uso registro del administrador



Figura 34. *Caso de uso registro del administrador*

Fuente: *Elaboración propia*

Tabla 15.

Descripción del administrador

CASO DE USO : INICIAR SESIÓN	
Actor(Es)	Personal (Administrador del Sistema)

Descripción	En este caso de uso se muestra como un usuario ingresa al sistema.
Propósito	Solo se permite el acceso a personas autorizadas
Prerrequisitos	El empleado debe ser un empleado vigente de la institución y estar registrarse al sistema para registrar usuario.

	Eventos Del Actor	Eventos Del Sistema
Flujo de Eventos	<p>1. el administrador ingresa su nombre de usuario y contraseña propia para el uso del sistema.</p> <p>2. el sistema verifica que exista sus datos (nombre y contraseña) en el sistema (logrease).</p> <p>3. El administrador tiene el visto bueno para administrar el sistema (inscribir usuario de viveros).</p>	<p>1. el sistema verifica el nombre y su contraseña, y si son incorrectos este mandara un mensaje de error.</p>
Post condición	El administrador será reconocido por el sistema bajo una previa validación de sus datos.	

Fuente: Elaboración propia

3.3.3 Caso de uso: registro de usuarios de viveros

Una vez teniendo una verificación de los documentos necesarios se realiza su registro correspondiente del usuario, sus empleados y todos los datos correspondientes al vivero para su posterior guardado en la base de datos.

Figura 3.2: Caso de uso registro de usuarios

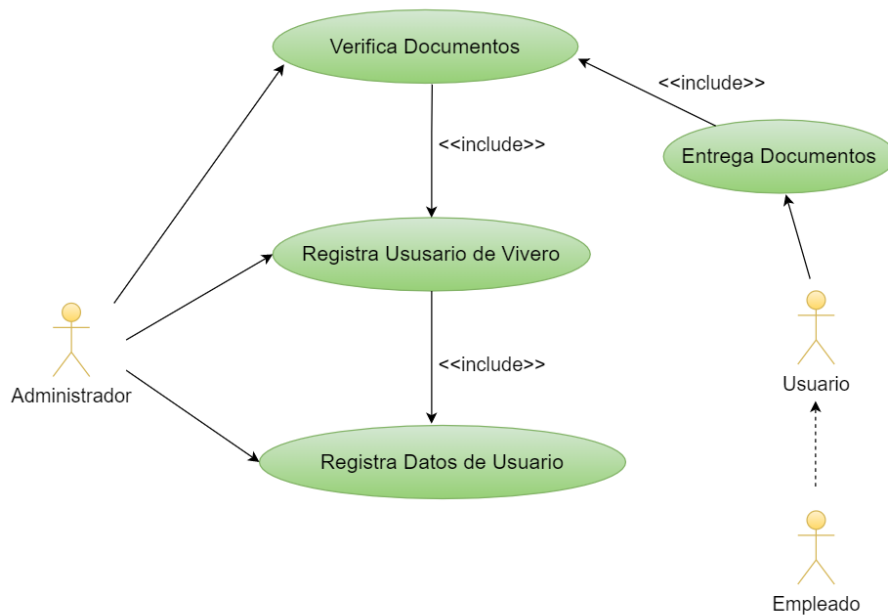


Figura 35. *Caso de uso registro de usuarios*

Fuente: Elaboración propia

Tabla 16.

Descripción del registro de usuarios

CASO DE USO : REGISTRO DE USUARIOS

Actor(es)	Personal (Administrador del Sistema, Usuarios)
Descripción	En este caso de uso se muestra como se realiza el registro de un usuario de viveros, sus datos complementarios (nombre, apellidos, cedula de identidad, dirección, correo electrónico, teléfono,).
Propósito	Obtener y registrar datos de los dueños de viveros existentes en cada regione.
Prerrequisitos	El usuario de los viveros debe tener los documentos y certificaciones necesarios para su registro.

Eventos Del Actor

Eventos Del Sistema

Flujo de Eventos	<ol style="list-style-type: none"> 1. el usuario entrega documentos y certificados para su respectivo registro. 2. el administrador verifica los documentos para su registro. 3. el administrador registra al usuario del vivero. 5. el administrador registra datos complementarios del usuario. 	<ol style="list-style-type: none"> 1. Inicia del sistema. 2. el sistema muestra el formulario de registro. 3. se registrar un nuevo usuario de vivero 4. se puede ver un listado de usuarios.
Post condición	El usuario del vivero debe presentar todos los documentos verídicos.	

Fuente: *Elaboración propia*

3.3.4 Caso de uso: registro de seguimientos

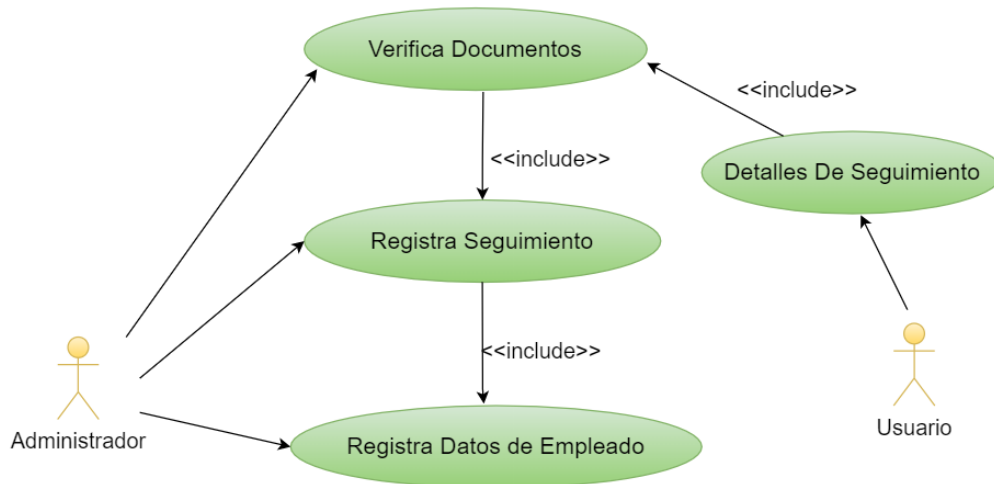


Figura 36. Caso de uso registro de seguimientos
Fuente: *Elaboración propia*

Tabla 17.
Descripción del registro seguimientos

CASO DE USO : REGISTRO DE SEGUIMIENTOS					
Actor(es)	Personal (Administrador del Sistema, Empleados)				
Descripción	En este caso de uso se muestra como se realiza el registro de los detalles de un viveros, sus datos complementarios como empleados, tipo de especies (nombre, genero tipo de empleado).				
Propósito	Obtener y registrar datos de los seguimientos para tener un total de tipo de trabajadores en un vivero existente en cada región.				
Prerrequisitos	Tener en cuenta el tipo y cantidad de empleado que consta el vivero para su registro.				
	<table border="0" style="width: 100%;"> <thead> <tr> <th style="width: 50%; text-align: center;">Eventos Del Actor</th> <th style="width: 50%; text-align: center;">Eventos Del Sistema</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;"> <ol style="list-style-type: none"> 1. el usuario entrega documentos para su respectivo registro. 2. el administrador verifica los documentos. 3. el administrador registra el tipo y totalidad de empleados que trabajan en el vivero. </td> <td style="vertical-align: top;"> <ol style="list-style-type: none"> 1. Inicia del sistema. 2. el sistema muestra el formulario de registro. 3. se registrar un nuevo seguimiento de un vivero 4. se despliega un listado de detalles. </td> </tr> </tbody> </table>	Eventos Del Actor	Eventos Del Sistema	<ol style="list-style-type: none"> 1. el usuario entrega documentos para su respectivo registro. 2. el administrador verifica los documentos. 3. el administrador registra el tipo y totalidad de empleados que trabajan en el vivero. 	<ol style="list-style-type: none"> 1. Inicia del sistema. 2. el sistema muestra el formulario de registro. 3. se registrar un nuevo seguimiento de un vivero 4. se despliega un listado de detalles.
Eventos Del Actor	Eventos Del Sistema				
<ol style="list-style-type: none"> 1. el usuario entrega documentos para su respectivo registro. 2. el administrador verifica los documentos. 3. el administrador registra el tipo y totalidad de empleados que trabajan en el vivero. 	<ol style="list-style-type: none"> 1. Inicia del sistema. 2. el sistema muestra el formulario de registro. 3. se registrar un nuevo seguimiento de un vivero 4. se despliega un listado de detalles. 				
Flujo de Eventos					

5. el administrador registra datos complementarios del seguimiento.

Post condición

Los empleados de los viveros deben presentar todos los documentos necesarios para su registro.

Fuente: *Elaboración propia*

3.3.5 Caso de uso: registro de vivero

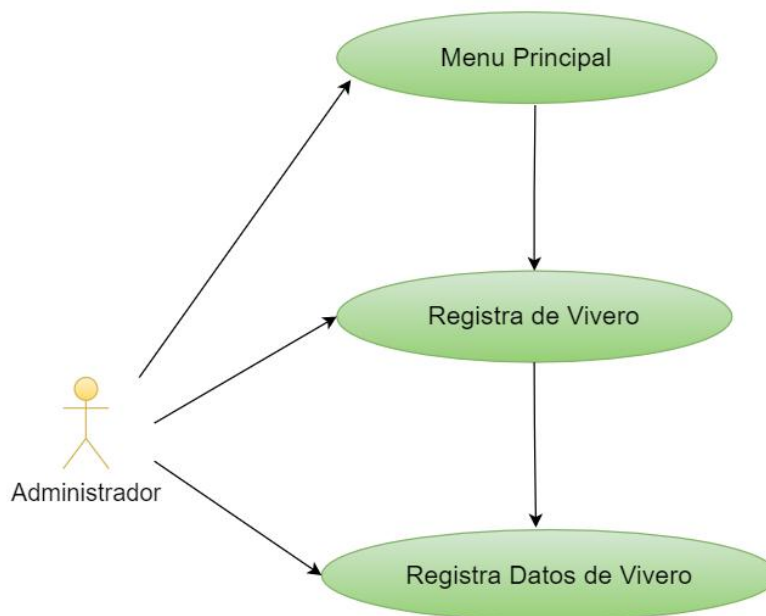


Tabla 18. *Caso de uso registro de vivero*

Fuente: *Elaboración propia*

Tabla 19.

Descripción de registro de viveros

CASO DE USO : REGISTRO DE VIVEROS

Actor(es)

Personal (Administrador del Sistema, Vivero)

Descripción En este caso de uso se muestra como se realiza el registro de los viveros, sus datos complementarios como (nombre del vivero, suministro de agua, tipo de semilla, superficie, Fase y tipo de vivero, producción, longitud, latitud).

Propósito Obtener y registrar datos de los viveros existentes en cada región del departamento.

Prerrequisitos Los viveros deben constar con una localización para su registro.

	Eventos Del Actor	Eventos Del Sistema
Flujo de Eventos	1. el dueño del vivero debe certificar la existencia del mismo para su respectivo registro. 3. el administrador registra el vivero. 5. el administrador registra datos complementarios del vivero.	1. Inicia del sistema. 2. el sistema muestra el formulario de registro de viveros. 3. se registrar un nuevo vivero 4. se puede ver un listado de los viveros registrados.

Post condición Los dueños de los viveros deben proporcionar todo los datos para su registro.

Fuente: *Elaboración propia*

3.3.6 Caso de uso: registro de especies

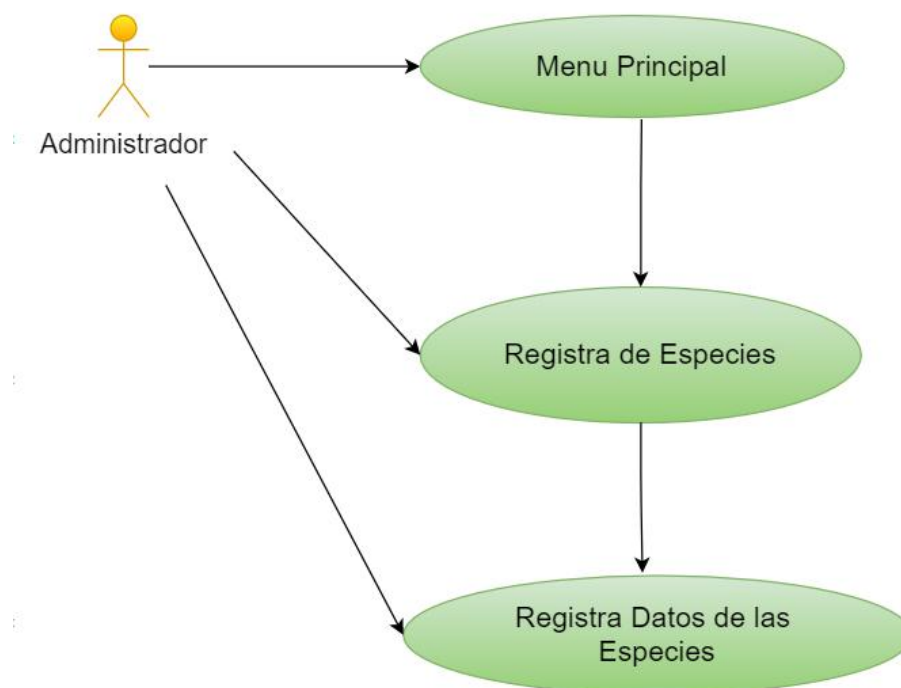


Figura 37. *Caso de uso registro de especies*

Fuente: *Elaboración propia*

Tabla 20.

Descripción de registro de especies

CASO DE USO : REGISTRO DE LAS ESPECIES	
Actor(es)	Personal (Administrador del Sistema, Especies)
Descripción	En este caso de uso se muestra como se realiza el registro de las especies de cultivo de un vivero, y sus datos como (nombre de la especie, tipo de la especie de cultivo).
Propósito	Obtener y registrar datos de tipo de cultivo que tiene el cada vivero existente registrado en cada región del departamento.
Prerrequisitos	Los viveros deben constar con un tipo especificado de cultivo.

	Eventos Del Actor	Eventos Del Sistema
Flujo de Eventos	<ol style="list-style-type: none"> 1. el dueño del vivero debe especificar el tipo cultivo tienen para su respectivo registro. 3. el administrador registra las características de la especie del vivero. 5. el administrador registra especie del vivero. 	<ol style="list-style-type: none"> 1. Inicia del sistema. 2. el sistema muestra el formulario de registro de especies. 3. se registra una nueva especie.
Post condición		Los dueños de los viveros deben dar datos de la especie para su registro.

Fuente: Elaboración propia

3.3.7 Caso de uso: registro de plagas del vivero

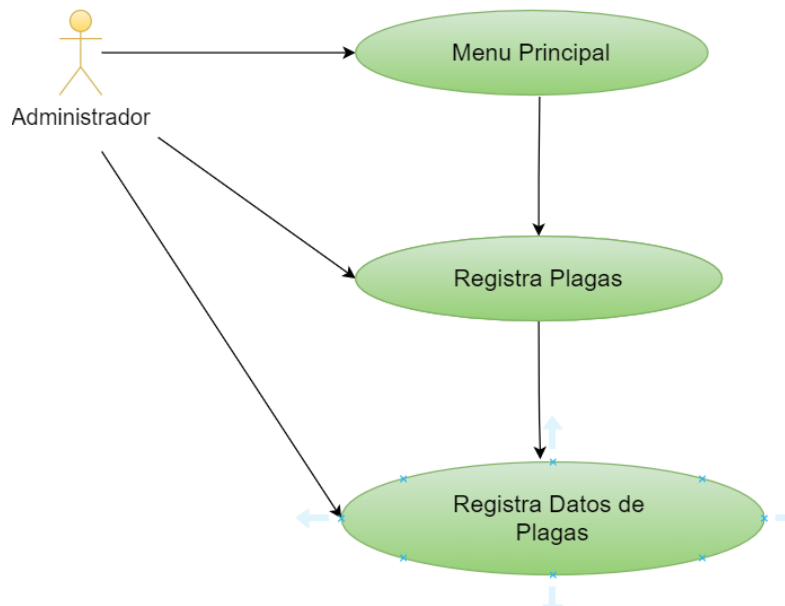


Figura 38. Caso de uso registro de plagas
Fuente: Elaboración propia

Tabla 21.
Descripción de registro de plagas

CASO DE USO : REGISTRO TIPO DE PLAGAS									
Actor(es)	Personal (Administrador del Sistema, Plagas)								
Descripción	En este caso de uso se muestra como se realiza el registro de los tipos de plaga que pueda presentar un cultivo de un vivero, y sus datos como (nombre de la plaga, tipo de plaga).								
Propósito	Obtener y registrar datos de tipo de plagas que pueda tener una plantación existente en cada vivero.								
Prerrequisitos	Los dueños deben constatar que tipo de plaga pueden tener.								
	<table border="0" style="width: 100%;"> <thead> <tr> <th style="width: 50%; text-align: center;">Eventos Del Actor</th> <th style="width: 50%; text-align: center;">Eventos Del Sistema</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;">1. el dueño del vivero debe especificar el tipo de plaga que puedan encontrar, para su respectivo registro.</td> <td style="vertical-align: top;">1. Inicia del sistema.</td> </tr> <tr> <td style="vertical-align: top;">3. el administrador registra las características de la plaga del vivero.</td> <td style="vertical-align: top;">2. el sistema muestra el formulario de registro de plagas. 3. se registra una nueva plaga.</td> </tr> <tr> <td style="vertical-align: top;">5. el administrador registra plaga del vivero.</td> <td></td> </tr> </tbody> </table>	Eventos Del Actor	Eventos Del Sistema	1. el dueño del vivero debe especificar el tipo de plaga que puedan encontrar, para su respectivo registro.	1. Inicia del sistema.	3. el administrador registra las características de la plaga del vivero.	2. el sistema muestra el formulario de registro de plagas. 3. se registra una nueva plaga.	5. el administrador registra plaga del vivero.	
Eventos Del Actor	Eventos Del Sistema								
1. el dueño del vivero debe especificar el tipo de plaga que puedan encontrar, para su respectivo registro.	1. Inicia del sistema.								
3. el administrador registra las características de la plaga del vivero.	2. el sistema muestra el formulario de registro de plagas. 3. se registra una nueva plaga.								
5. el administrador registra plaga del vivero.									
Flujo de Eventos									
Post condición	Los dueños de los viveros deben de reconocer las plagas potenciales para su registro.								

Fuente: *Elaboración propia*

3.3.8 Caso de uso: registro de localización del vivero

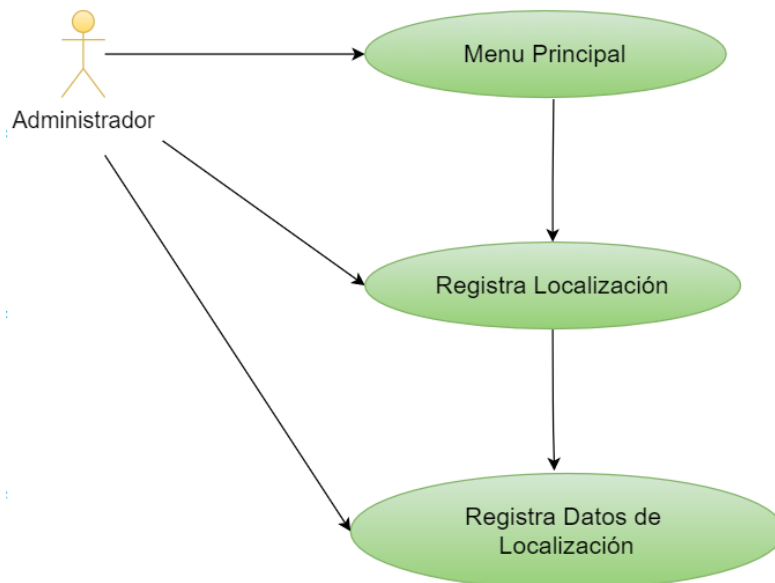


Figura 39. Caso de uso registro de localización

Fuente: *Elaboración propia*

Tabla 22.

Descripción de registro de localización

CASO DE USO : REGISTRO LOCALIZACIÓN DE VIVEROS	
Actor(es)	Personal (Administrador del Sistema, localización)
Descripción	En este caso de uso se muestra como se realiza el registro de la localización referencial del cultivo de un vivero, y sus datos como (departamento, provincia, municipio).
Propósito	Obtener y registrar datos referenciales de su ubicación y existencia de cada vivero.
Prerrequisitos	Los dueños de los viveros deben constar con área donde se ubique el vivero para su georreferencia.
Eventos Del Actor	Eventos Del Sistema

Flujo de Eventos	<ol style="list-style-type: none"> 1. el dueño del vivero debe especificar la localización del mismo, para su respectivo registro. 3. el administrador registra la ubicación georeferencial del vivero. 5. el administrador registra la ubicación geográfica del vivero. 	<ol style="list-style-type: none"> 1. Inicia del sistema. 2. el sistema muestra el formulario de registro de localización. 3. se registra una nueva localización.
Post condición	<p>Los dueños de los viveros deben dar un área (terreno) para su vivero para su registro.</p>	

Fuente: Elaboración propia

3.3.9 Caso de uso: registro de entidad financiera

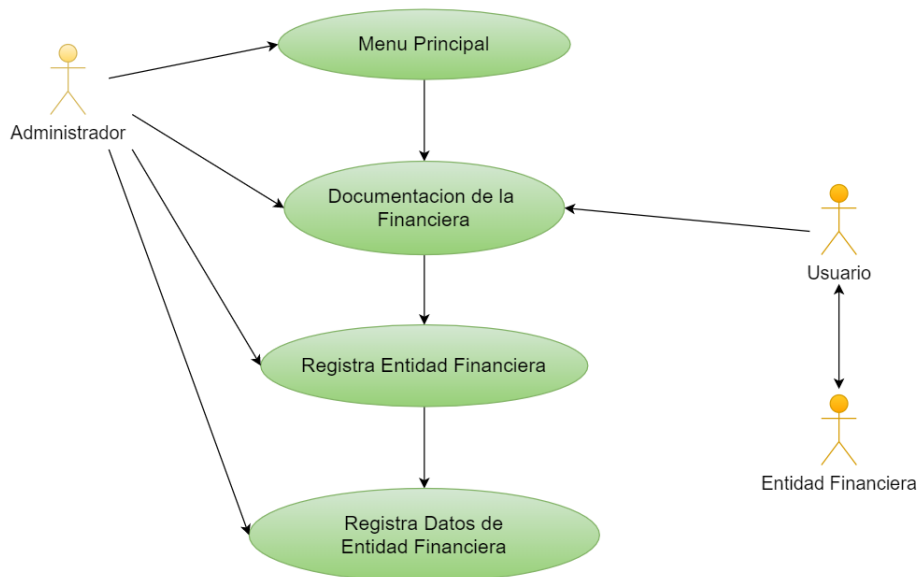


Figura 40. Caso de uso registro entidad financiera

Fuente: Elaboración propia

Tabla 23.
Descripción de registro entidad financiera

CASO DE USO : REGISTRO LA ENTIDAD FINANCIERA					
Actor(es)	Personal (Administrador del Sistema, financiador)				
Descripción	En este caso de uso se muestra como se realiza el registro de la entidad financiadora con la que cuenta un vivero como ser (nombre del financiador, nombre de la entidad financiadora y otros).				
Propósito	Obtener y registrar datos de los financiadores con la que cuenta un vivero.				
Prerrequisitos	Los dueños de viveros deben citar si cuentan con una entidad financiera para la creación de sus viveros.				
	<table border="0" style="width: 100%;"> <thead> <tr> <th style="width: 50%; text-align: center;">Eventos Del Actor</th> <th style="width: 50%; text-align: center;">Eventos Del Sistema</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;"> <p>1. el dueño del vivero debe especificar debe presentar documentación de los financiadores, para su respectivo registro.</p> <p>3. el administrador registra las características de la entidad financiadora.</p> <p>5. el administrador registra a los financiadores.</p> </td> <td style="vertical-align: top;"> <p>1. Inicia del sistema.</p> <p>2. el sistema muestra el formulario de registro de la entidad financiadora.</p> <p>3. se registra una nueva entidad financiadora.</p> </td> </tr> </tbody> </table>	Eventos Del Actor	Eventos Del Sistema	<p>1. el dueño del vivero debe especificar debe presentar documentación de los financiadores, para su respectivo registro.</p> <p>3. el administrador registra las características de la entidad financiadora.</p> <p>5. el administrador registra a los financiadores.</p>	<p>1. Inicia del sistema.</p> <p>2. el sistema muestra el formulario de registro de la entidad financiadora.</p> <p>3. se registra una nueva entidad financiadora.</p>
Eventos Del Actor	Eventos Del Sistema				
<p>1. el dueño del vivero debe especificar debe presentar documentación de los financiadores, para su respectivo registro.</p> <p>3. el administrador registra las características de la entidad financiadora.</p> <p>5. el administrador registra a los financiadores.</p>	<p>1. Inicia del sistema.</p> <p>2. el sistema muestra el formulario de registro de la entidad financiadora.</p> <p>3. se registra una nueva entidad financiadora.</p>				
Flujo de Eventos					
Post condición	Los dueños de los viveros deben al menos tener una entidad financiadora para su registro.				

Fuente: *Elaboración propia*

3.3.10 Diagrama de paquetes

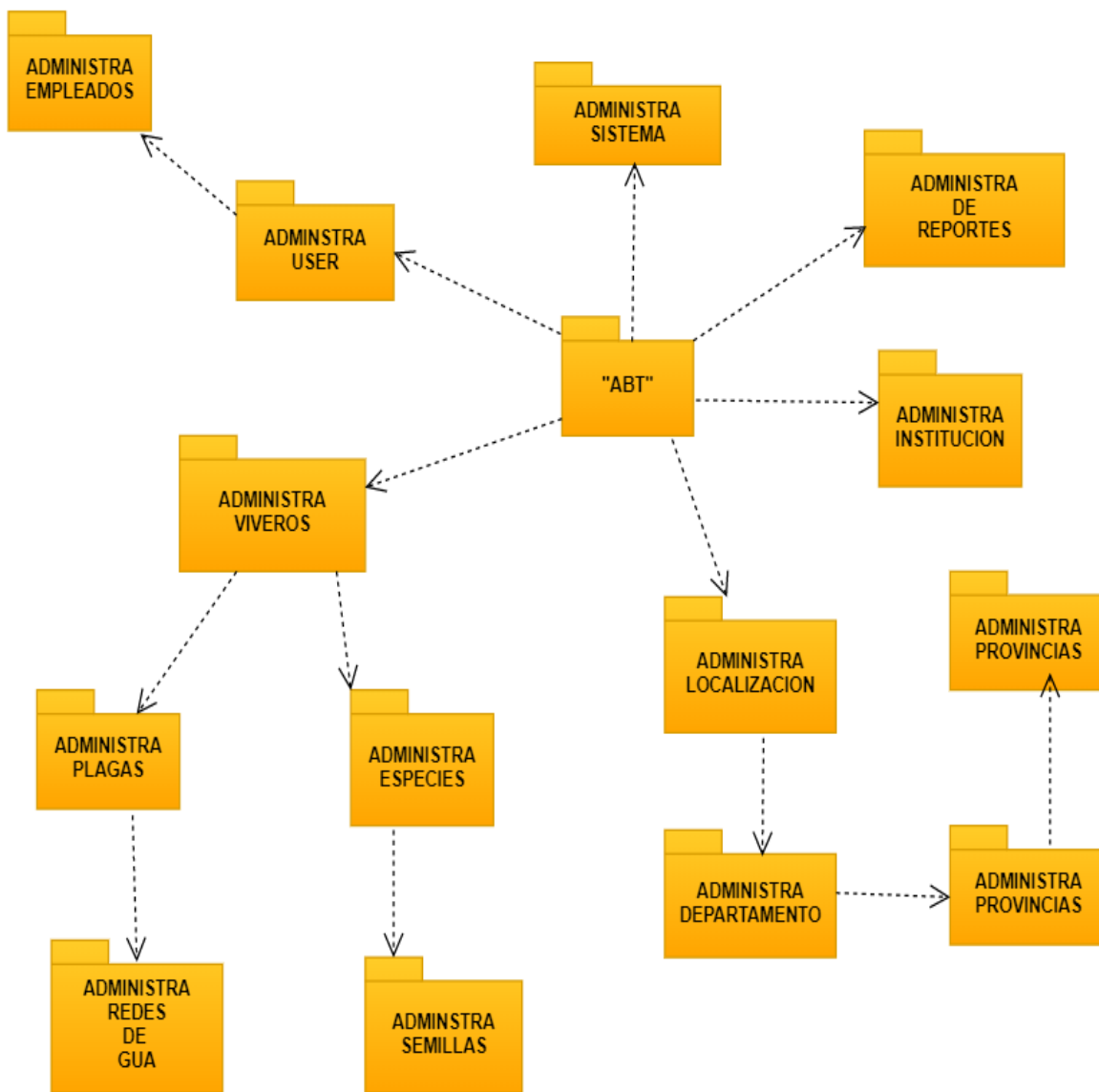


Figura 41. Diagrama de paquetes de ABT

Fuente: Elaboración propia

3.3.11 Diagrama de clases



Figura 42. Diagrama de clases
Fuente: Elaboración propia

3.3.12 Diagrama físico

El diagrama de diseño físico, describe cada una de las clases de dominio del sistema y la relación con cada una de las clases.

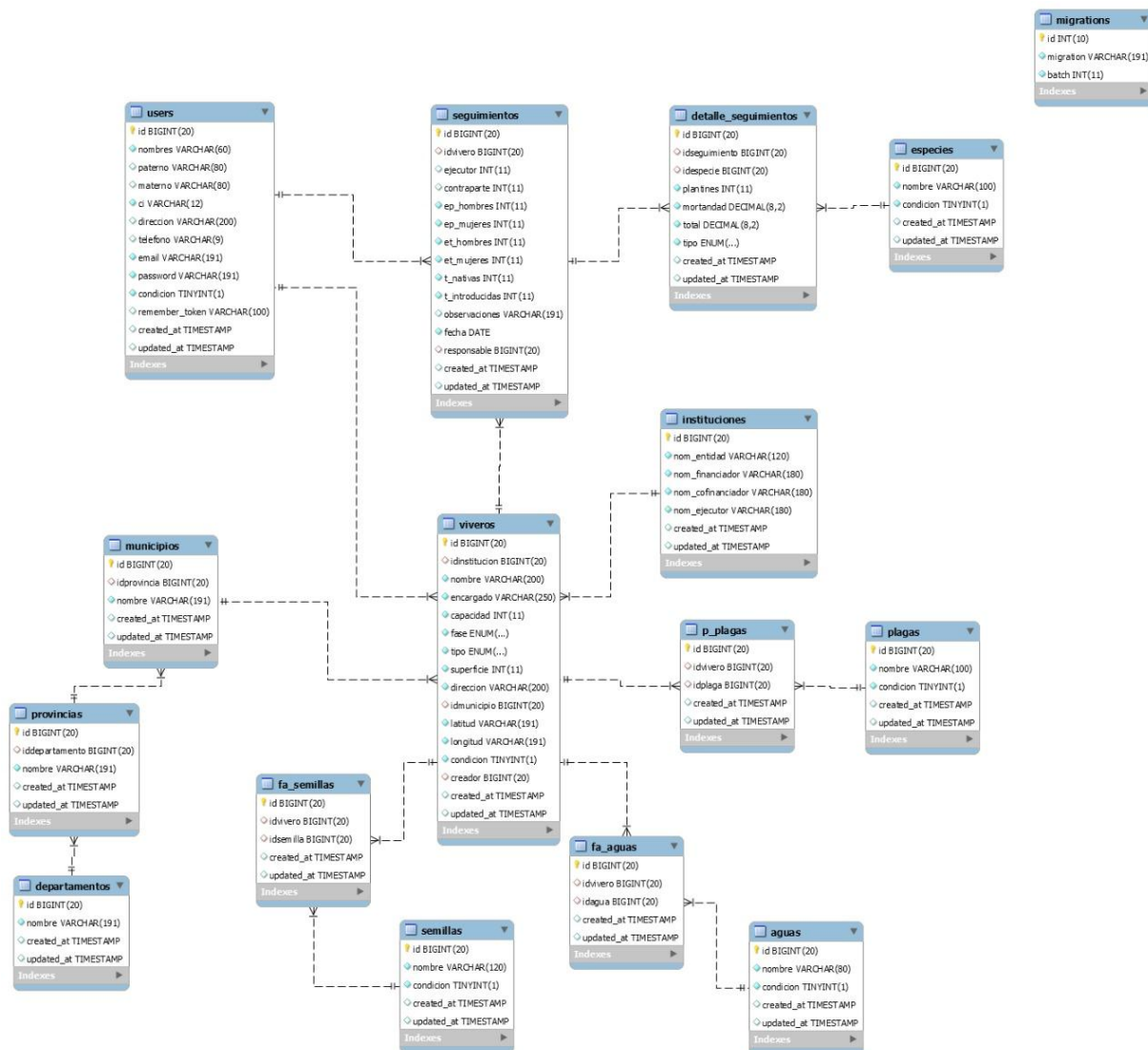


Figura 43. Diagrama de diseño físico
Fuente: Elaboración propia

3.4 MODELO

3.4.1 Modelo de navegación

En el diagrama navegación general. Se describe la función de cada actividad del sistema en forma general y como es su navegar.

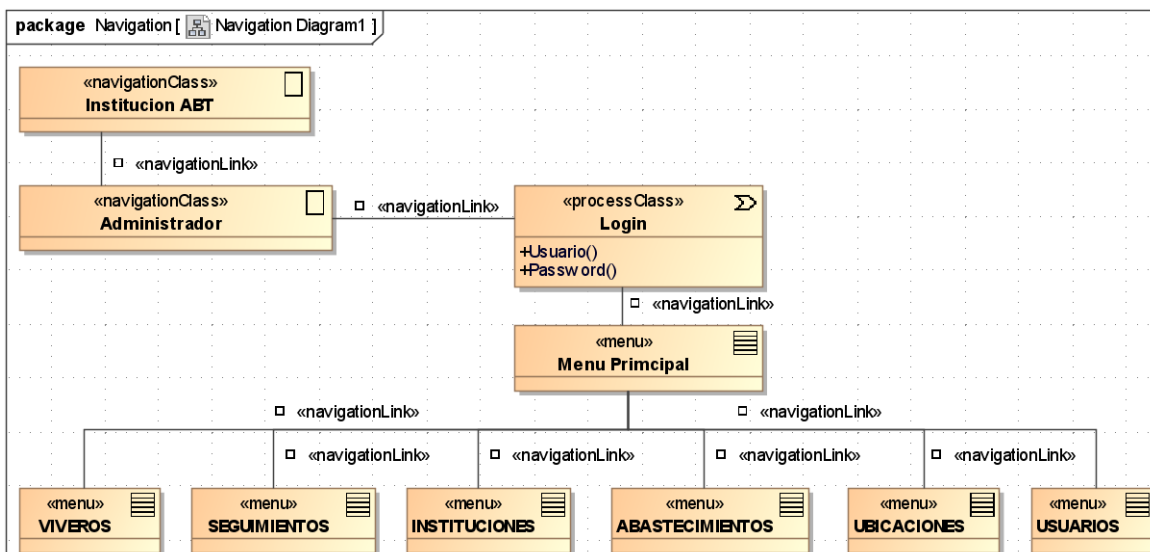


Figura 44. Modelo de Navegación Menú Principal

Fuente: Elaboración propia

3.4.1.1 Modelo de navegación de viveros

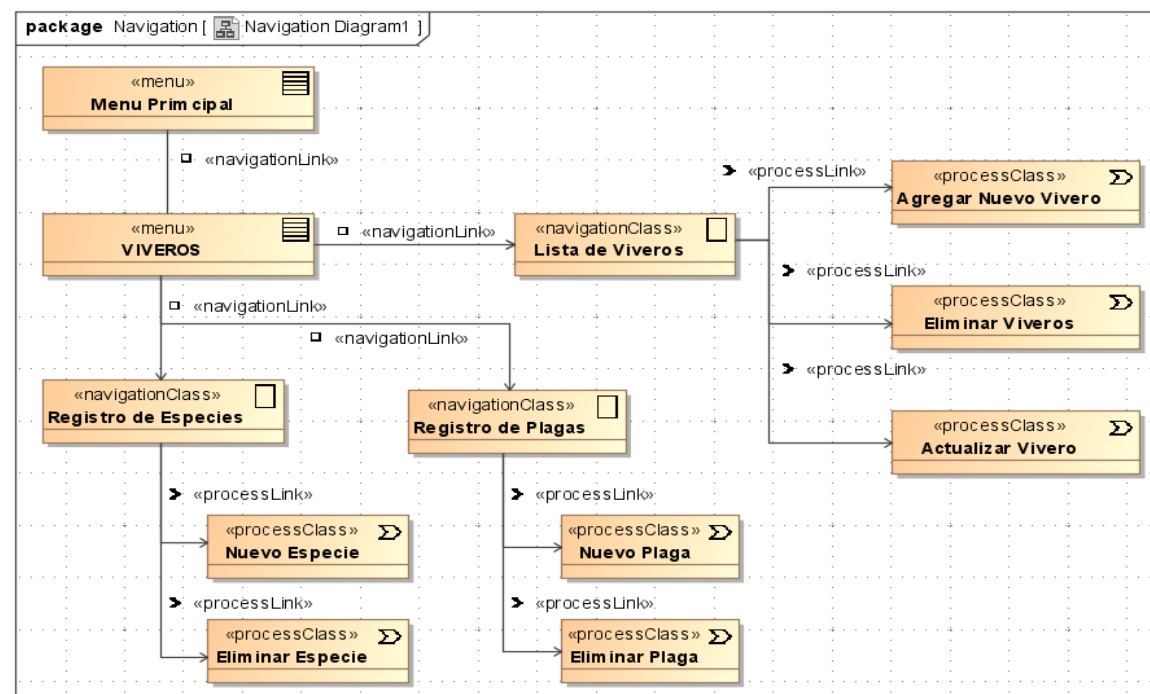


Figura 45. Modelo de Navegación de Viveros

Fuente: Elaboración propia

3.4.1.2 Modelo de navegación de seguimiento

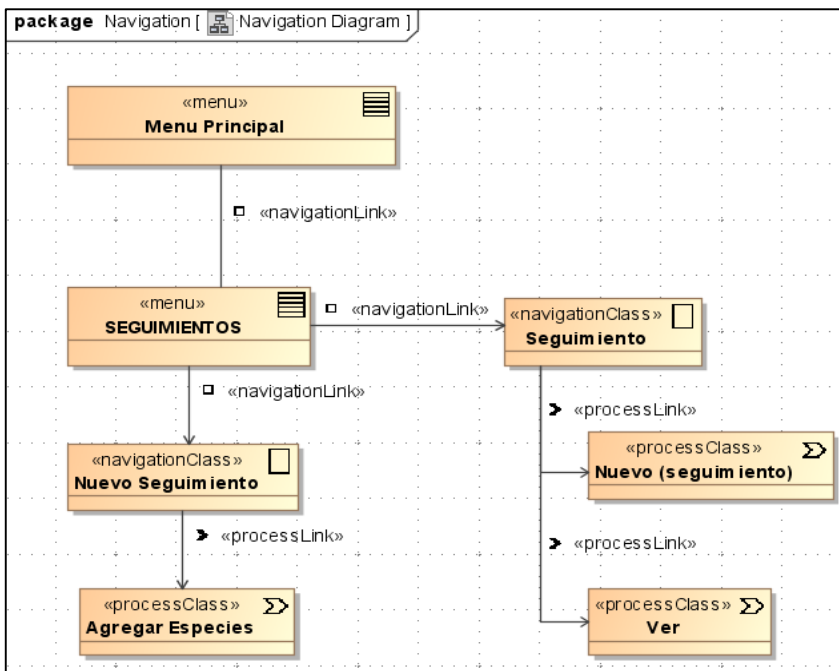


Figura 46. Modelo de Navegación de Seguimiento
Fuente: Elaboración propia

3.4.1.3 Modelo de navegación de institución

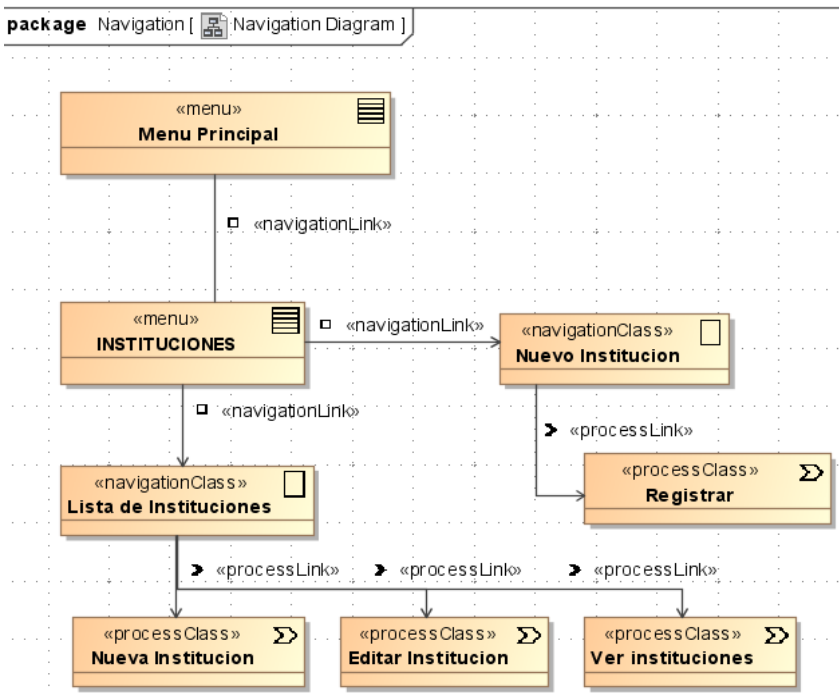


Figura 47. Modelo de Navegación de Institución
Fuente: Elaboración propia

3.4.1.4 Modelo de navegación de abastecimiento

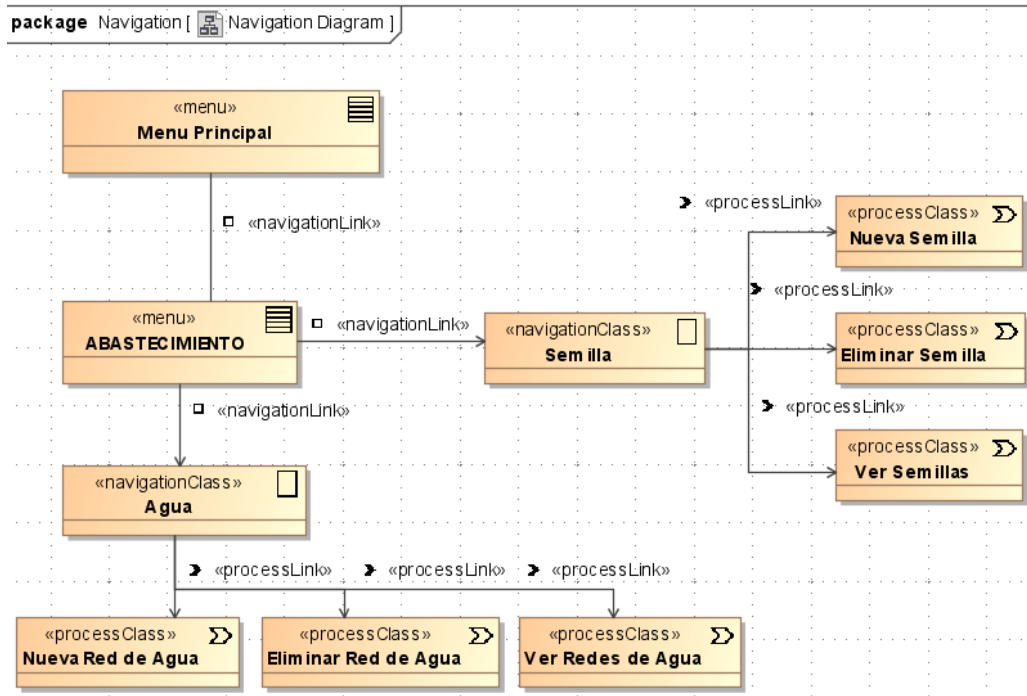


Figura 48. Modelo de Navegación de Abastecimiento

Fuente: Elaboración propia

3.4.1.5 Modelo de navegación de ubicación

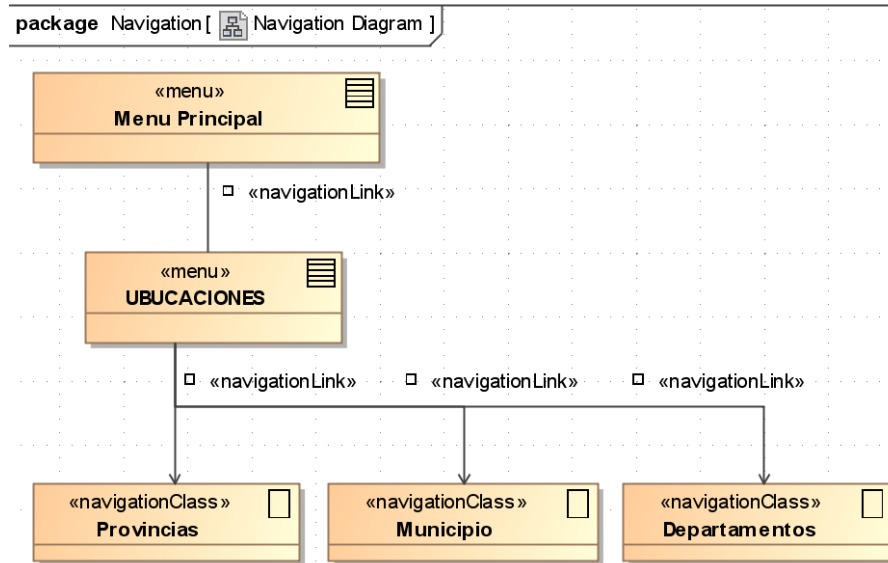


Figura 49. Modelo de Navegación Ubicación

Fuente: Elaboración propia

3.4.1.6 Modelo de navegación de usuario

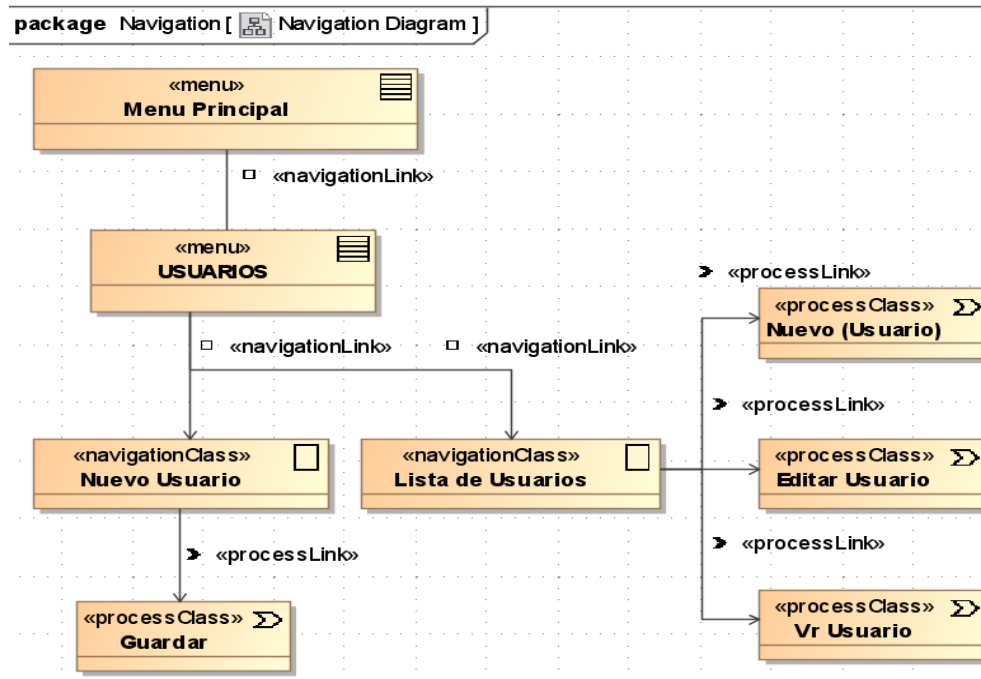


Figura 50. Modelo de Navegación Usuario
Fuente: Elaboración propia

3.4.2 Modelo de presentación

3.4.2.1 Modelo de presentación registro del administrador

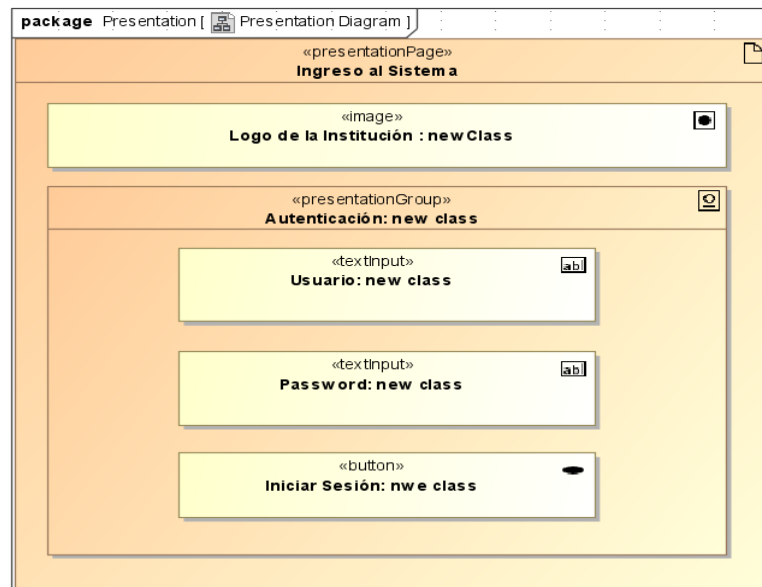


Figura 51. Modelo de Presentación registro del administrador
Fuente: Elaboración propia

3.4.2.2 Modelo de presentación página principal

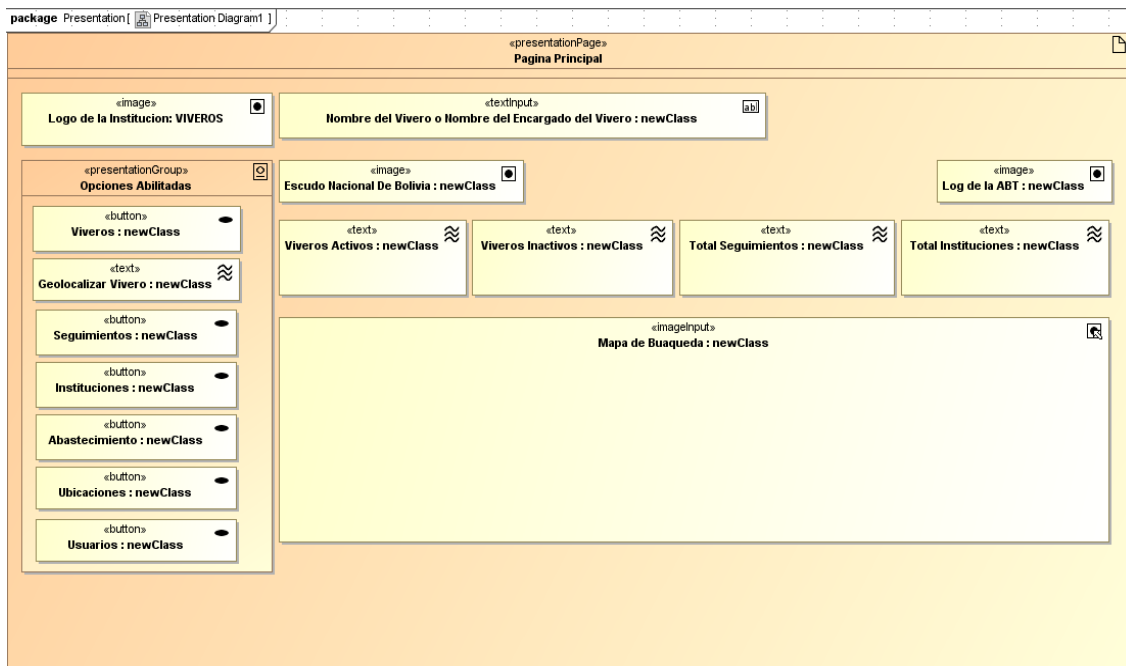


Figura 52. Modelo de Presentación Página Principal
Fuente: Elaboración propia

3.4.2.3 Modelo de presentación listado de viveros

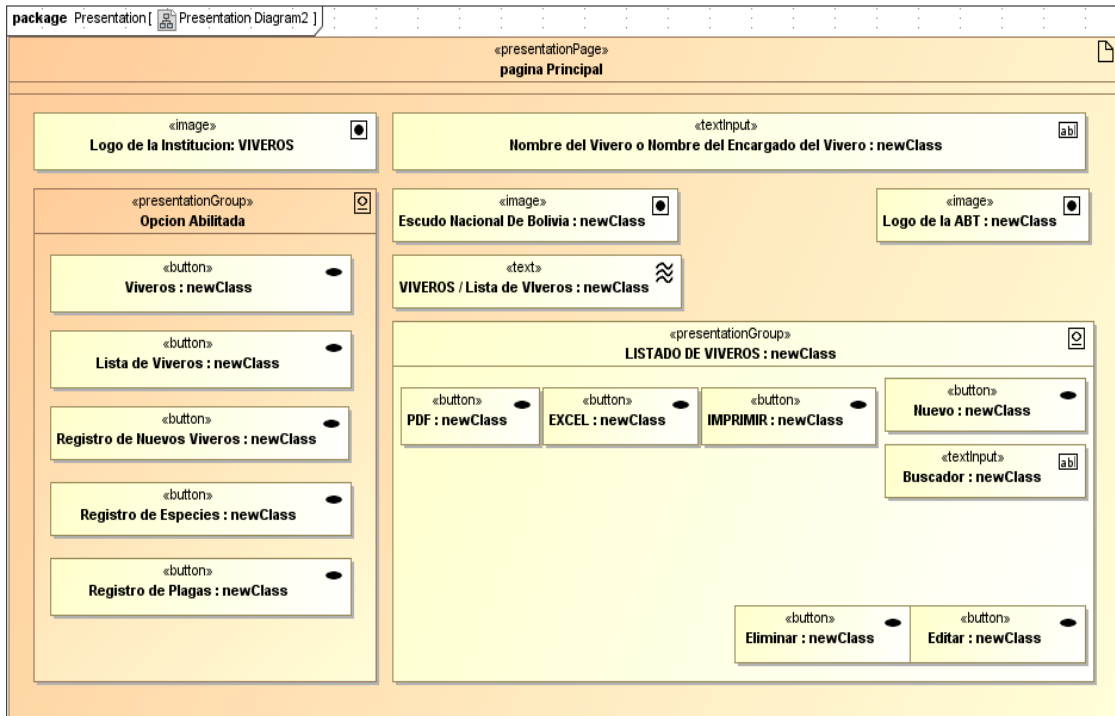


Figura 53. Modelo de Presentación Listado de Viveros
Fuente: Elaboración propia

3.4.2.4 Modelo de presentación listado de especies

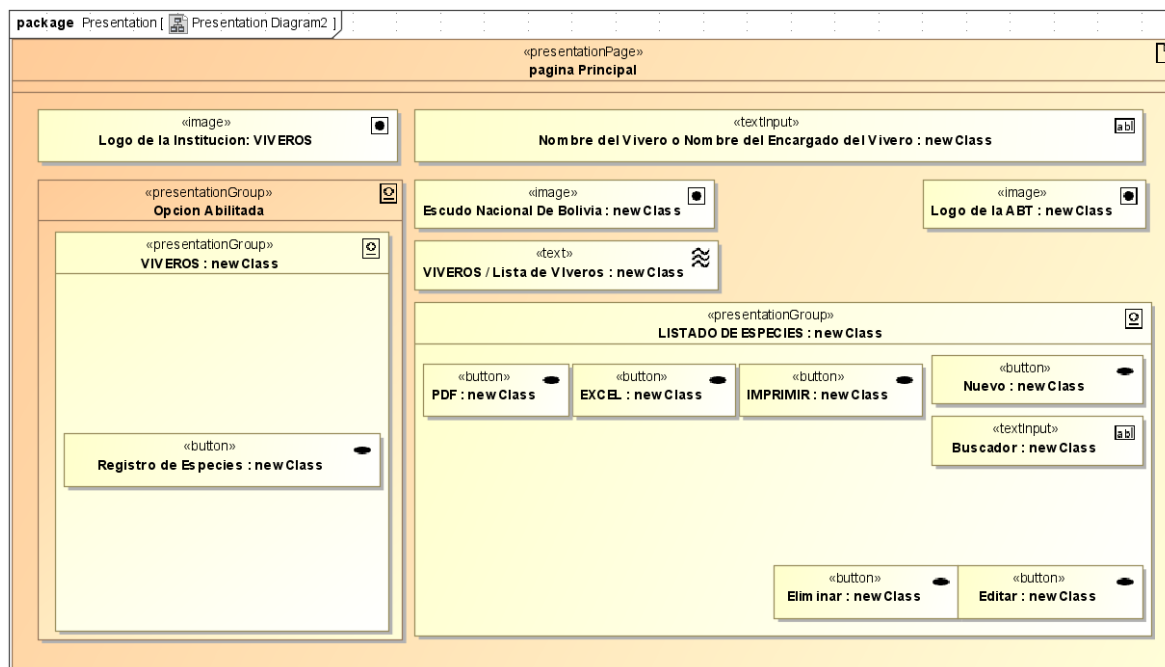


Figura 54. Modelo de Presentación Listado de Especies

Fuente: Elaboración propia

3.4.2.5 Modelo de presentación listado de plagas

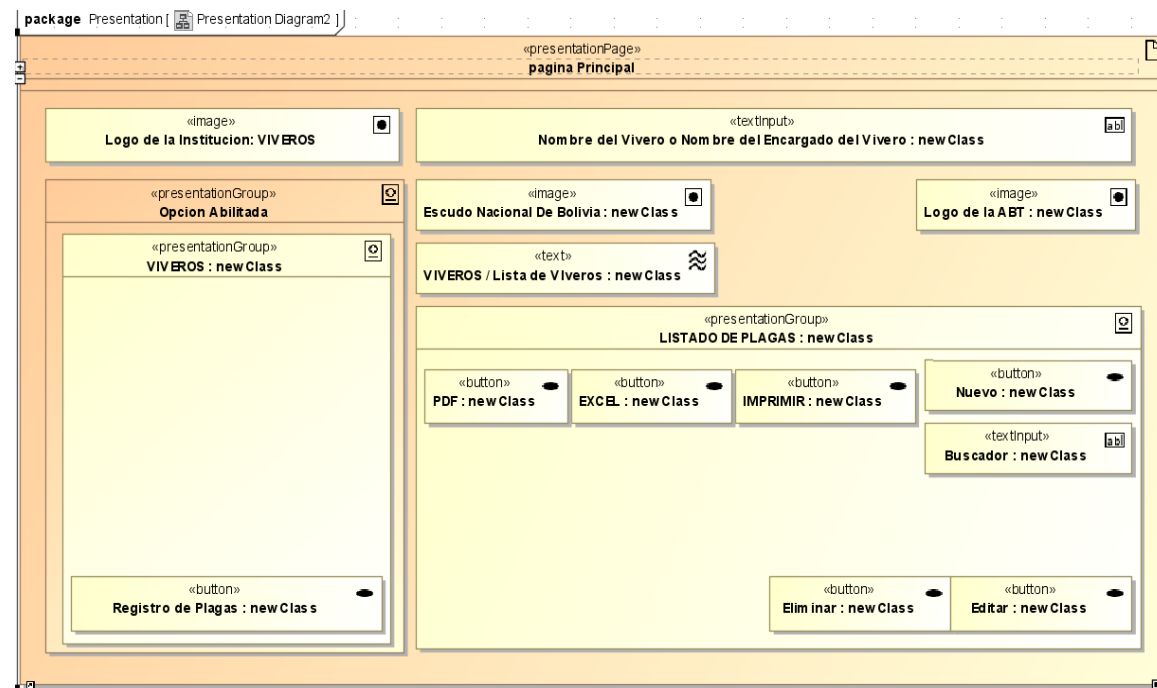


Figura 55. Modelo de Presentación Listado de Plagas

Fuente: Elaboración propia

3.4.2.6 Modelo de presentación listado de seguimientos

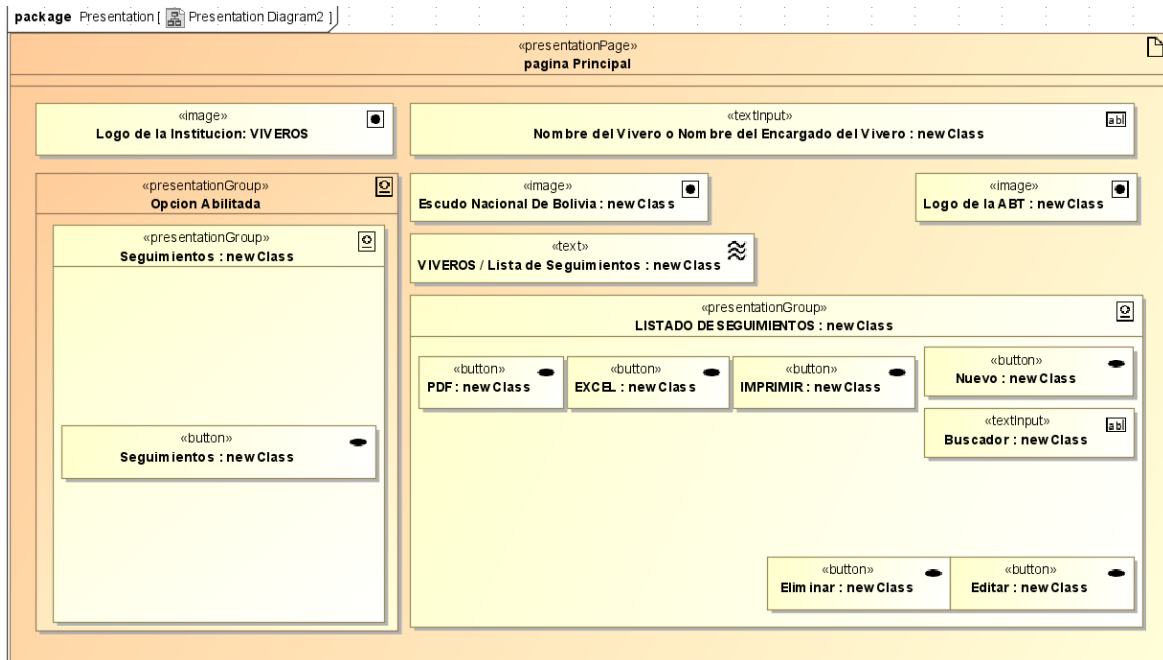


Figura 56. Modelo de Presentación Listado de Seguimientos

Fuente: Elaboración propia

3.4.2.7 Modelo de presentación listado de instituciones

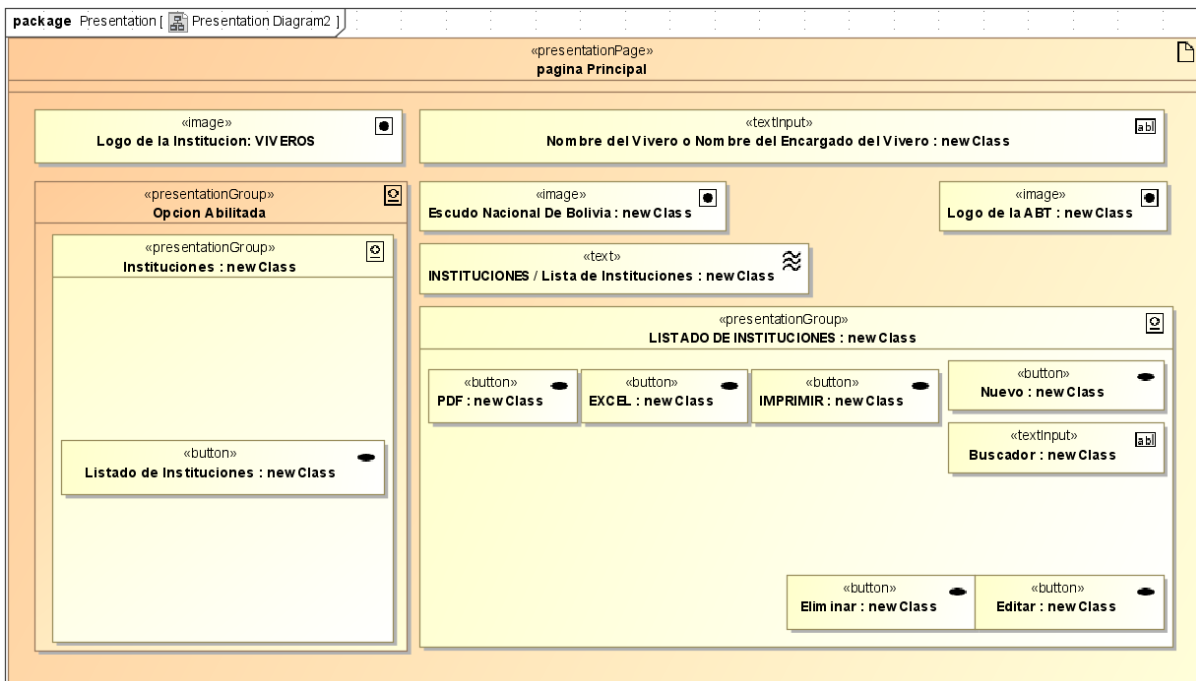


Figura 57. Modelo de Presentación Listado de Instituciones

Fuente: Elaboración propia

3.4.2.8 Modelo de presentación listado de abastecimiento de agua

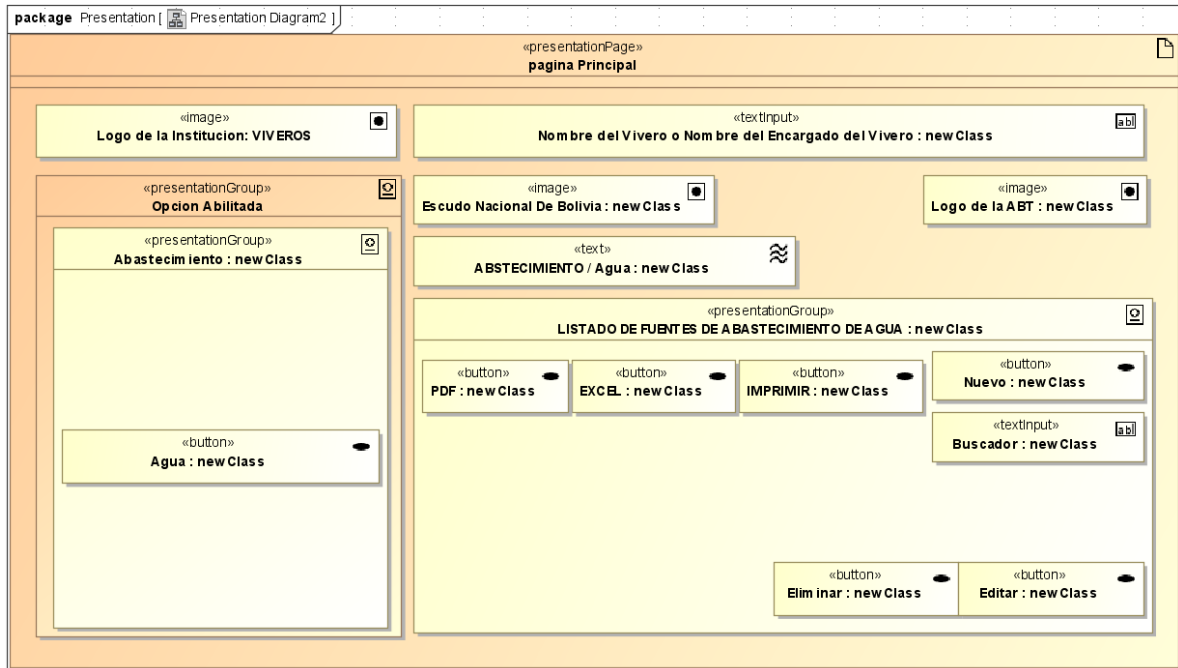


Figura 58. Modelo de Presentación Listado de Abastecimiento de Agua
Fuente: Elaboración propia

3.4.2.9 Modelo de presentación listado de abastecimiento de semilla

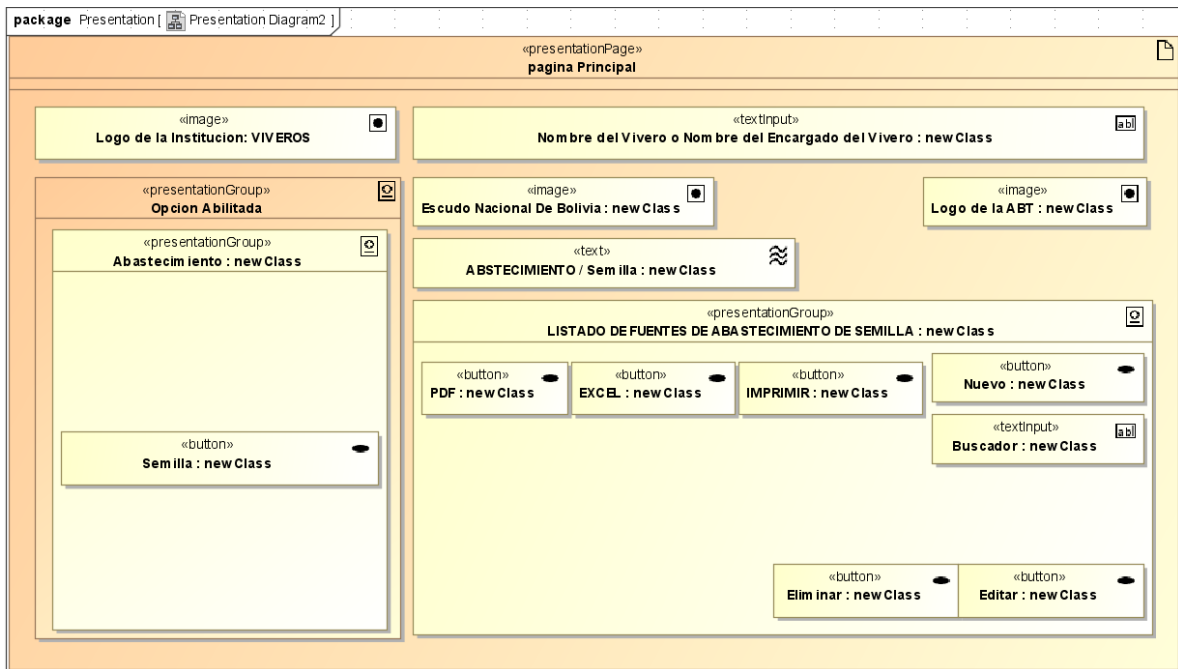


Figura 59. Modelo de Presentación Listado de Abastecimiento de Semilla
Fuente: Elaboración propia

3.4.2.10 Modelo de presentación listado de departamentos

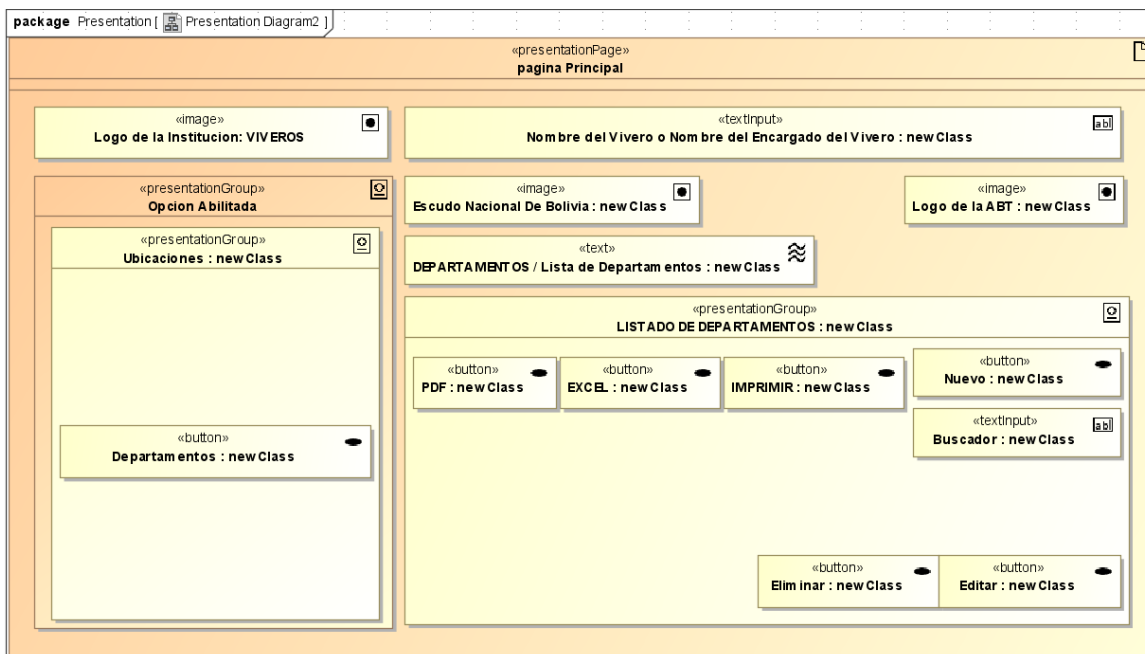


Figura 60. Modelo de Presentación Listado de Departamentos

Fuente: Elaboración propia

3.4.2.11 Modelo de presentación listado de usuarios

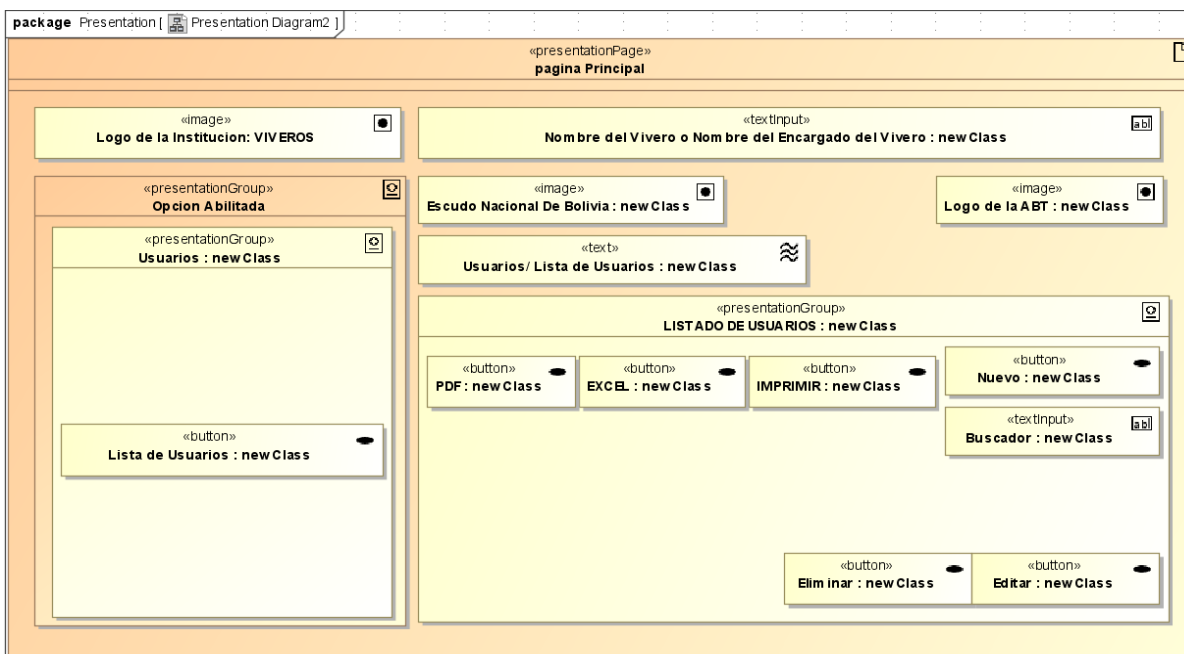


Figura 61. Modelo de Presentación Listado de Usuarios

Fuente: Elaboración propia

3.5 IMPLEMENTACION DEL SISTEMA

En esta etapa veremos cada una de las interfaces presentadas en la fase de diseno, pero esta vez ya en de una manera visual al usuario para su manejo.

3.5.1 Interfaz de inicio de sesión

Una vez que el usuario se encuentre en cualquier navegador, el ingresa al link de la página se encontrara en la vista de inicio de sesión donde registrándose como usuario y con una contraseña.



Figura 62. Inicio de sesión
Fuente: *Elaboración propia*

3.5.2 Interfaz de menú de navegación

Una vez que ya iniciamos sesión, veremos la pantalla principal que es el menú de navegación del sistema donde el administrador realizara el registro de los viveros.

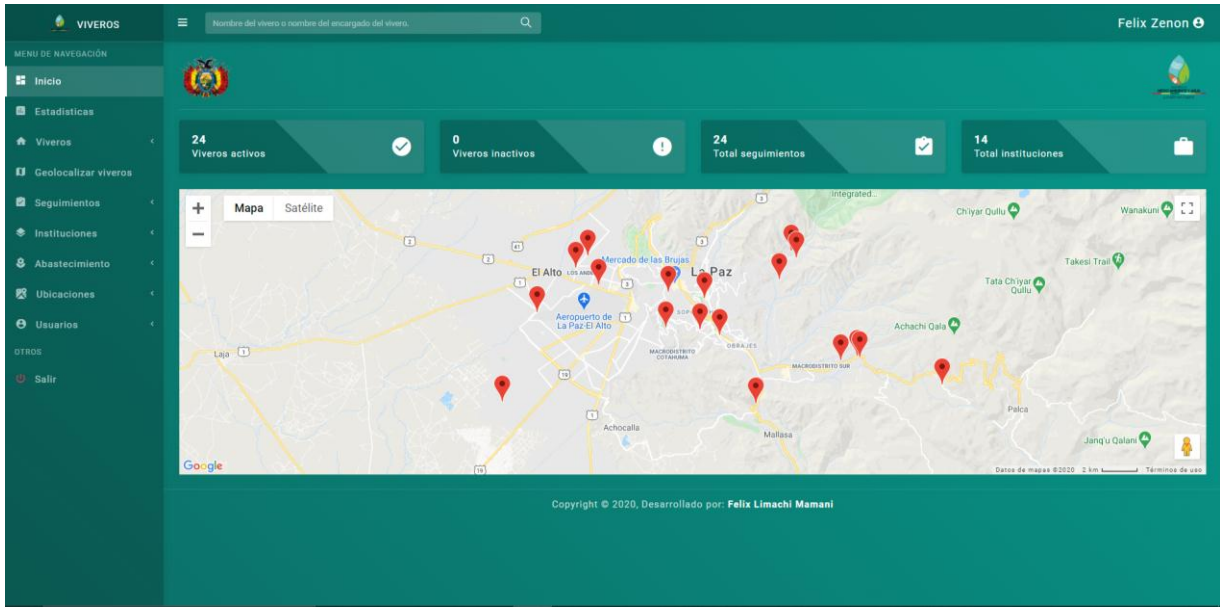


Figura 63. pantalla de principal (menú de navegación)
 Fuente: Elaboración propia

3.5.3 Interfaz de estadísticas de los viveros

En esta interfaz se puede apreciar la relación entre los viveros de plantas nativas e introducidas registradas por año

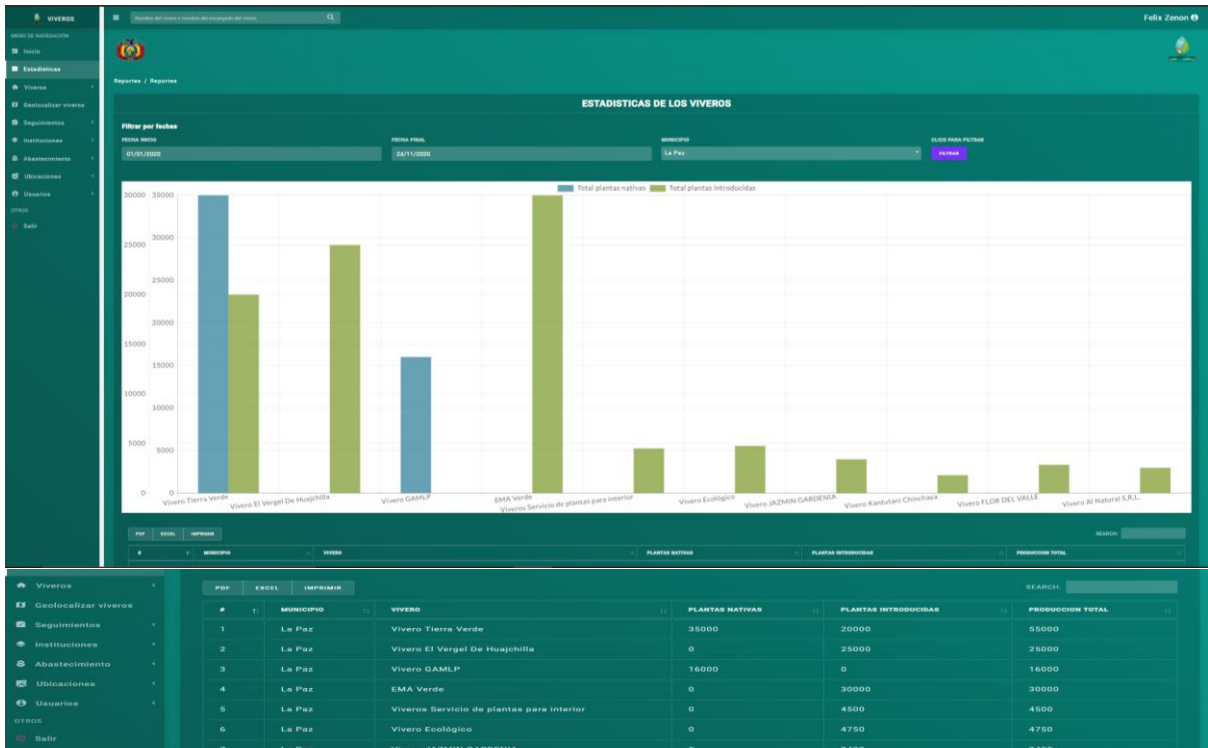


Figura 64. Pantalla de estadísticas
 Fuente: Elaboración propia

3.5.4 Interfaz de registro de viveros

En la interfaz siguiente podemos ver la pantalla de registro de un vivero donde el administrador registrara algunas de las características como registro de especie, registro de tipo de especie, registro de tipo de plaga y listado de viveros.

REGISTRO DE VIVEROS

INSTITUCION A CARGO DEL VIVERO: Banco Ganadero

MUNICIPIO EN QUE SE ENCUENTRA EL VIVERO: Sucre

NOMBRE DEL VIVERO: Nombre del vivero

CODIGO DEL VIVERO: 2020024

NOMBRE DEL PROPIETARIO: Nombre del propietario del vivero

CAPACIDAD PRODUCTIVA: Capacidad productiva del vivero

TIPO DE VIVERO: Temporal

FASE DE VIVERO: Produccion

DIRECCION DEL VIVERO: Direccion del vivero

SUPERFICIE DEL VIVERO (EN M2): Superficie del vivero

FUENTES DE ABASTECIMIENTO DE AGUA:

- RIOS
- POZOS
- ATAJO
- RED DE DISTRIBUCION
- LAGOS
- MANANTIALES
- FERRO CEMENTO
- EMBALSES
- CISTERNA
- GALERIA FILTRANTE

UBICACION REFERENCIAL (ARRASTRE EL PUNTERO):

BUSCAR: CEJA puente El Alto, Av 6 de Marzo, El Alto, Bolivia

Figura 65. Registro de viveros

Fuente: Elaboración propia

En la siguiente imagen podemos ver el registro y listado del tipo de especie y el nombre común de la especie.

VIVEROS / Registro tipos de especies

LISTADO DE TIPOS DE ESPECIES

PDF EXCEL IMPRIMIR

SEARCH: []

#	TIPO DE ESPECIE	CONDICION	ACCIONES
1	Árboles maderables	Activo	[e] [v]
2	Árboles ornamentales	Activo	[e] [v]
3	Arbustos	Activo	[e] [v]
4	Bonsáis	Activo	[e] [v]
5	Bonsáis Caducos	Activo	[e] [v]
6	Bonsáis Hojas anchas perennes	Activo	[e] [v]
7	Bonsáis Coníferas y pinos	Activo	[e] [v]
8	Bulbosas	Activo	[e] [v]
9	Cactus	Activo	[e] [v]
10	Crasas o Suculentas	Activo	[e] [v]

Mostrando 1 al 10 de 19 registros

Previous 1 2 Next

Copyright © 2020, Desarrollado por: Felix Limachi Mamani

Figura 66. Pantalla del listado de tipo de especies

Fuente: Elaboración propia

#	TIPO ESPECIE	NOMBRE DE ESPECIE	CONDICION	ACCIONES
1	Árboles maderables	Ajunao	Activo	[B] [V]
2	Árboles maderables	Algarrobilla	Activo	[B] [V]
3	Árboles maderables	Cedro	Activo	[B] [V]
4	Árboles maderables	Jorori	Activo	[B] [V]
5	Árboles maderables	Mara	Activo	[B] [V]
6	Árboles maderables	Sirari	Activo	[B] [V]
7	Árboles maderables	Serebó	Activo	[B] [V]
8	Árboles maderables	Tajibo Amarillo	Activo	[B] [V]
9	Árboles maderables	Tajibo Morado	Activo	[B] [V]
10	Árboles maderables	Tajibo Rosa y Fucsia	Activo	[B] [V]

Figura 67. Registro de especies
Fuente: *Elaboración propia*

En la vista siguiente nos muestra el listado de plagas registradas que puedan tener un vivero, así mismo acá se realiza el registro de una nueva plaga que no se encuentre en el listado.

#	NOMBRE	CONDICION	ACCIONES
1	Hongos	Activo	[B] [V]
2	Bacterias	Activo	[B] [V]
3	Virus	Activo	[B] [V]
4	Insecto	Activo	[B] [V]
5	Acaros	Activo	[B] [V]
6	Plantas parasitas	Activo	[B] [V]

Figura 68. Registro de plagas
Fuente: *Elaboración propia*

3.5.5 Interfaz de seguimiento

En esta interfaz nos muestra una vista de registro de empleados sean estos permanentes o temporales, así también un menú para el registro referente a la especie. También una vez terminado el registrado, en esta interfaz se genera un documento pdf con toda la información recolectada referente de los viveros.



Figura 69. interfaz de seguimiento
Fuente: *Elaboración propia*

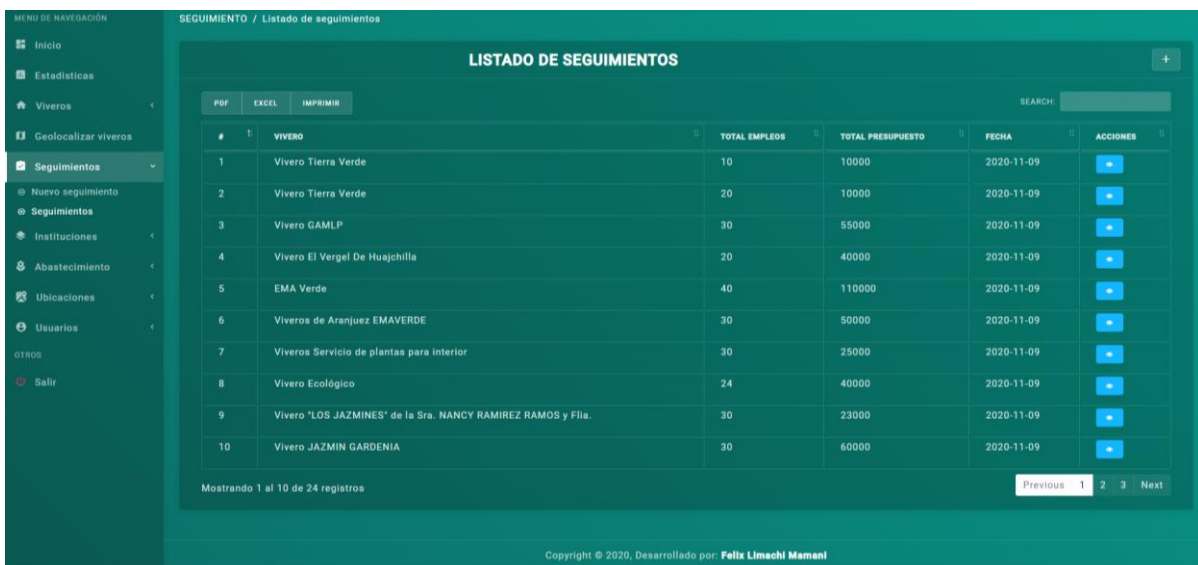




Figura 70. Listado de seguimiento
Fuente: *Elaboración propia*

Una vez que tengamos toda la información referente a los viveros se generara un documento pdf con información suficiente para los dueños de viveros y así mismo desde acá se puede proporcionar información de las características de los viveros.

Guía de Relevamiento de Datos e Información-Fase Viveros

**PROGRAMA NACIONAL DE FORESTACION Y REFORESTACION
FASE VIVEROS**

NOMBRE FINANCIADOR	NOMBRE COFINANCIADOR	NOMBRE EJECUTOR	CODIGO (SISTEMA)				
Armando Contreras	Felix Robles	Ramiro Kanktuta	20201				
INSTITUCION A CARGO	SUPERFICIE VIVERO	PRESUPUESTO EJECUTOR	PRESUPUESTO CONTRAPARTE				
Fundacion Vida	100	15000	1000				
DATOS DE UBICACION:							
Departamento:	Provincia:	Municipio:	Dirección/Lugar:				
La Paz	Murillo	Achocalla	Francisco Bedrgal entre Puente Union				
Nombre del vivero:	Nombre propietario:	Capacidad productiva:	Fecha seguimiento:				
EMAVEVERDE	Zenon Mamani	200	2020-08-24				
EMPLIDOS PERMANENTES:	EMPLIDOS TEMPORALES:	FASE:	TIPO:				
Total	0	Total	10	Produccion		Temporal	
Hombres	0	Hombres	10	Fortalecimiento		Permanente	X
Mujeres	0	Mujeres	0	Construccion	X		

DATOS DEL VIVERO:

Fuentes de abastecimiento de agua:

Ferro cemento

Croquis referencial



Fuentes de abastecimiento de agua:

Rodales semilleros

PRODUCCION DE ESPECIES NATIVAS				PRODUCCION DE ESPECIES INTRODUCIDAS							
Nombre especie	N plantines	% mortandad	Total	Nombre especie	N plantines	% mortandad	Total				
				Gazania	100	10.00	90.00				
TOTAL NATIVAS				TOTAL INTRODUCIDAS				TOTAL PRODUCCION			
0				90				90			

Plagas presentes en el vivero:

Insecto

Responsable del llenado:	Observaciones:
Felix Zenon Limachi Mamani	

Firma.....

Figura 71. Documento pdf de vivero
Fuente: *Elaboración propia*

3.5.6 Interfaz de registro de institución

En esta interfaz se realiza el registro de una nueva institución con la que contara un vivero como facilitador de estos. Al momento de realizar el registro de un vivero se debe de seleccionar la institución existente en el listado, con la que contara como facilitador, en caso de no exista la

institución se debe de registrar primero la institución y luego empezar con el registro de un nuevo vivero

The screenshot shows a web interface for registering a new institution. The page title is 'REGISTRO DE INSTITUCION'. It features a sidebar menu on the left with options like 'Inicio', 'Estadísticas', 'Viveros', 'Geolocalizar viveros', 'Seguimientos', 'Instituciones', 'Nueva Institución', 'Lista de Instituciones', 'Abastecimiento', 'Ubicaciones', 'Usuarios', and 'Salir'. The main content area contains a form with the following fields:

- NOMBRE DE LA INSTITUCION:** Ingrese nombre de la institución
- NOMBRE DEL FINANCIADOR:** Ingrese nombre del financiador
- NOMBRE DEL COFINANCIADOR:** Ingrese nombre del cofinanciador
- NOMBRE DEL EJECUTOR:** Ingrese nombre del ejecutor

A 'REGISTRAR' button is located at the bottom right of the form. The footer indicates 'Copyright © 2020, Desarrollado por: Felix Limachi Mamani'.

Figura 72. Registro de institución
Fuente: *Elaboración propia*

Esta es la interfaz de las instituciones registradas como facilitadores de viveros.

The screenshot shows a web interface for listing registered institutions. The page title is 'LISTADO DE INSTITUCIONES'. It features a sidebar menu on the left with options like 'Inicio', 'Estadísticas', 'Viveros', 'Geolocalizar viveros', 'Seguimientos', 'Instituciones', 'Nueva Institución', 'Lista de Instituciones', 'Abastecimiento', 'Ubicaciones', 'Usuarios', and 'Salir'. The main content area contains a table with the following data:

#	NOMBRE	FINANCIADOR	COFINANCIADOR	EJECUTOR	ACCIONES
1	Nus+eva Institucion	Moises Quispe	Marco Valle	Alan Villa	[Edit] [Delete]
2	Banco De Agricultura	Teodocito Tico	Marco Valle	Alberto Beto	[Edit] [Delete]
3	Financiera Ecologica 2	Moises Quispe	Marco Valle	Alan Villa	[Edit] [Delete]
4	Financiera Agil	Alberto Leon	Jose Quispe	Lucas Milan	[Edit] [Delete]
5	Banco Agricola Ganadera	Alberto Leon	Teodocio Talle	Lucas Milan	[Edit] [Delete]
6	Banco Agricola	Luis Mamani	Saul Calle	Alberto Beto	[Edit] [Delete]
7	Financiera Para El Desarrollo Agricola	Braulio Quispe	Matias Maco	Alberto Beto	[Edit] [Delete]
8	Banco Financiero Ecologico	Teodocito Tico	Marco Valle	Lucas Milan	[Edit] [Delete]
9	Vida Verde 2	Moises Quispe	Marco Valle	Joel Segundo	[Edit] [Delete]
10	Banco Semillero	Sergio Calle	Sergio Mamani	Saul Sales	[Edit] [Delete]

At the bottom of the table, it says 'Mostrando 1 al 10 de 14 registros'. There are also 'Previous', '1', '2', and 'Next' pagination controls. The footer indicates 'Copyright © 2020, Desarrollado por: Felix Limachi Mamani'.

Figura 73. Listado de instituciones
Fuente: *Elaboración propia*

3.5.7 Interfaz de registro de abastecimiento

Esta es la interfaz donde se muestra un listado de fuentes de abastecimientos de agua, así también se puede registrar uno nuevo en caso de no exista en la lista.

#	NOMBRE	CONDICION	ACCIONES
1	Rios	Activo	[Red] [Blue]
2	Lagos	Activo	[Red] [Blue]
3	Embalses	Activo	[Red] [Blue]
4	Pozos	Activo	[Red] [Blue]
5	Manantiales	Activo	[Red] [Blue]
6	Cisterna	Activo	[Red] [Blue]
7	Atajo	Activo	[Red] [Blue]
8	Ferro cemento	Activo	[Red] [Blue]
9	Galeria filtrante	Activo	[Red] [Blue]
10	Red de distribución	Activo	[Red] [Blue]

Figura 74. Listado y registro de fuentes de agua
Fuente: Elaboración propia

En esta interfaz como en la anterior se ve el listado y se puede registrar uno nuevo, en caso de que no exista, se lo registra.

#	NOMBRE	CONDICION	ACCIONES
1	Huertos semilleros	Activo	[Red] [Blue]
2	Rodales semilleros	Activo	[Red] [Blue]
3	Banco de semillas certificadas	Activo	[Red] [Blue]
4	Carlos el Semillero	Activo	[Red] [Blue]
5	Huertos semilleros 2	Activo	[Red] [Blue]
6	Huertos semilleros 3	Activo	[Red] [Blue]
7	Rodales semilleros2	Activo	[Red] [Blue]
8	Rodales semilleros3	Activo	[Red] [Blue]
9	Banco de semillas	Activo	[Red] [Blue]
10	Banco de semillas fortificadas	Activo	[Red] [Blue]

Figura 75. listado y registro de semillas
Fuente: Elaboración propia

3.5.8 Interfaz de departamentos

En esta interfaz nos muestra la cantidad de municipios y provincias de un departamento.



The screenshot shows a web application interface for 'VIVEROS'. The main content area is titled 'LISTADO DE DEPARTAMENTOS' and contains a table with the following data:

#	NOMBRE	CANT. PROVINCIAS	CANT. MUNICIPIOS	CANT. VIVEROS
1	Chuquisaca	10	33	5
2	La Paz	20	102	19
3	Cochabamba	16	0	0
4	Oruro	16	0	0
5	Potosí	16	0	0
6	Tarija	6	0	0
7	Santa Cruz	15	0	0
8	Beni	8	0	0
9	Pando	5	0	0

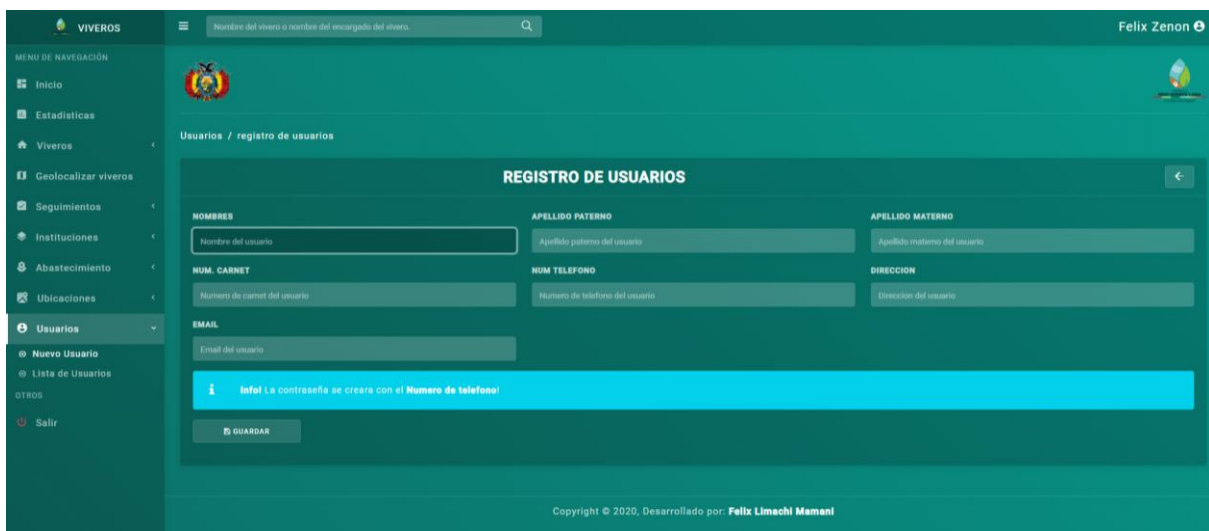
Below the table, it indicates 'Mostrando 1 al 9 de 9 registros' and has 'Previous' and 'Next' navigation buttons. The footer of the interface states 'Copyright © 2020, Desarrollado por: Felix Limechi Mamani'.

Figura 76. Listado de departamento

Fuente: Elaboración propia

3.5.9 Interfaz de registro de usuario

Esta interfaz es la última en donde el administrador puede crea, elimina un nuevo usuario, para que pueda acceder al sistema.



The screenshot shows a web application interface for 'VIVEROS' with the title 'REGISTRO DE USUARIOS'. The form contains the following fields:

- NOMBRES: Nombre del usuario
- APELLIDO PATERNO: Apellido paterno del usuario
- APELLIDO MATERNO: Apellido materno del usuario
- NUM. CARNET: Numero de carnet del usuario
- NUM TELEFONO: Numero de telefono del usuario
- DIRECCION: Direccion del usuario
- EMAIL: Email del usuario

Below the form, there is a blue information box that reads: 'Info! La contraseña se creara con el Numero de telefono!'. A 'GUARDAR' button is located at the bottom of the form. The footer of the interface states 'Copyright © 2020, Desarrollado por: Felix Limechi Mamani'.

Figura 77. Registro de un nuevo usuario

Fuente: Elaboración propia

#	NOMBRES	APELLIDOS	TELEFONO	DIRECCION	CONDICION	ACCIONES
1	Felix Zenon	Limachi Mamani	65191901	en casa pues we	Activo	[+]

Figura 78. *Listado de usuarios*

Fuente: *Elaboración propia*

3.6 METRICAS DE CALIDAD

Existen muchos autores de definen la Calidad del Software, Pressman define a la calidad del Software como “la concordancia con los requisitos funcionales y de rendimientos explícitamente establecidos, estándares de desarrollo explícitamente documentados y características implícitas que se espera de todo software desarrollado profesionalmente”.

El objetivo principal de esta técnica es alcanzar la calidad necesaria para satisfacer las necesidades del cliente. La calidad según esta norma ISO 9126 puede ser medida de acuerdo a los factores:

- Usabilidad
- Funcionalidad
- Confiabilidad
- Mantenibilidad
- Portabilidad

3.6.1 Usabilidad

La usabilidad consiste de un conjunto de atributos que permite evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema, es decir realizar una serie de preguntas que permiten ver cuán sencillo, fácil de aprender y manejar es para los usuarios. Esta comprensión por parte de los usuarios con relación al sistema evalúa los siguientes pasos

- Entendimiento
- Aprendizaje

- Operabilidad
- Atracción
- Conformidad de uso

En la siguiente tabla se observa estos criterios en niveles de porcentajes a los que llegó el sistema en cuanto a su comprensibilidad, para el usuario, y posteriormente se da el porcentaje final de usabilidad del sistema. Se realizó una encuesta al usuario final sobre el manejo, la comprensión y la facilidad de aprender el sistema para medir la usabilidad según la siguiente tabla.

Tabla 24.
Cuadro de Usabilidad

N°	PREGUNTAS	RESPUESTAS		PORCENTAJE
		SI (1 - 10)	NO (1 - 10)	
1	¿Puede utilizar con facilidad el sistema?	8	1	1.14
2	¿Puede controlar operaciones que el sistema solicita?	6	1	0.86
3	¿Las respuestas del sistema son complicadas?	5	2	0.71
4	¿El sistema permitió la retroalimentación de información?	5	1	0.71
5	¿El sistema cuenta con interfaces agradables a la vista?	5	1	0.71
6	¿La respuesta del sistema es satisfactorio?	6	1	0.86
7	¿Los resultados que proporciona el sistema facilitan el trabajo?	8	1	1.14
PROMEDIO TOTAL				0.87

Fuente: *Elaboración propia*

$$U = (\text{sum } (x_i)/n) * 100$$

$$U = 0.87 * 100$$

$$U = 87$$

USABILIDAD = 87%

De acuerdo a los datos obtenidos en la tabla de usabilidad, se concluye que el sistema tiene una usabilidad del 87%. Esto indica que el usuario tiene una conformidad hacia el sistema y que este podrá coadyuvar de alguna manera a solucionar los problemas de la empresa planteados anteriormente.

3.6.2 Funcionalidad

La funcionalidad examina si el sistema satisface los requisitos funcionales esperados. El objetivo es revelar problemas y errores en lo que concierne a la funcionalidad del sistema y su conformidad al comportamiento, expresado o deseado por el usuario. Se cuantifica el tamaño y la complejidad del sistema en términos de las funciones de usuario, puede ser valorado mediante el Punto Función. Se basa en la contabilización de cinco parámetros los cuales se desarrollan a continuación:

- **Número de entradas de usuario**

Se refiere a cada entrada de control del usuario que proporciona diferentes datos al sistema.

- **Número de salidas de usuario**

Se refiere a cada salida de información que proporciona el sistema al usuario, entre estos están: informes, pantallas, mensajes de errores, entre otros.

- **Número de peticiones de usuario**

Una petición se define como una entrada interactiva que produce la generación de alguna respuesta de software en forma de salidas interactivas

- **Número de Archivos**

Se cuenta archivos maestro lógico, estos pueden ser: grupo lógico de datos que sean parte de una base de datos, o un archivo independiente.

- **Número de interfaces externas**

Se cuenta las interfaces legibles por la máquina que se utilizan para transmitir información a otro sistema.

Para calcular el Punto Función se utilizará la siguiente fórmula:

$$\text{Punto Funcion (PF)} = [\text{Total} * (X + (\text{Min} * \sum (Fi)))]$$

Donde:

PF: Medida de la funcionalidad.

X: Confiabilidad del proyecto, varía entre el 1% a 100%

Min (y): Error mínimo aceptable al de la complejidad, el margen de error es igual a 0.01.

Fi: Son los valores de ajuste de la complejidad, donde i=1 a i=14.

En la siguiente tabla se calcula el punto función, los cuales miden el software desde una perspectiva del usuario, dejando de lado los detalles de codificación.

Tabla 25.
Cuadro de Punto Fusión y sus Respectivas características y pesos

PARAMETROS	CANTIDAD	PESO	PESO	PESO	FACTOR DE PESO
		SIMPLE	MEDIO	COMPLEJO	
N.E.U.	24	3	4	6	96
N.S.U.	20	4	5	7	80
N.P.U.	10	3	4	6	40
N.F.	8	7	10	15	80
N.I.E.	0	5	7	10	0
TOTAL					296

Fuente: *Elaboración propia*

Donde:

N.E.U.: Es el número de entradas de usuario para proporcionar datos al software.

N.S.U.: Es el número de salidas que proporcionan al usuario información.

N.P.U.: Es el número de peticiones que se realizan al usuario como respuesta a una salida interactiva.

N.F.: Es el número de ficheros maestros que se realizan en la aplicación.

N.I.E.: Es el número de interfaces externas que se utilizan para la transmisión de la información.

Cada organización que utiliza métodos de puntos desarrolla criterios para determinar si una entrada es denominada simple o compleja. Los valores de Fi, se obtiene de los resultados de la siguiente tabla, bajo las ponderaciones descritos en la escala.

Tabla 26.
Cuadro de ajuste de complejidad (Sumatorias Σfi)

IMPORTANCIA	0	1	2	3	4	5	Fi
ESCALA	SIN IMPORTANCIA INCREMENTAL		MODERADO	NEDIO	SIGNIFICATIVO	ESENCIAL	
1.- ¿Requiere el sistema copias de seguridad y de recuperación fiable?					X		4
2.- ¿Se requiere comunicación de datos?					X		4
3. ¿Existe funciones de procesamiento distribuidos?				X			3
4. ¿Es critico el rendimiento?				X			3
5.- ¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?				X			3
6.- ¿Requiere el sistema entrada de datos interactiva?						X	5
7.- ¿Requiere el sistema entrada de datos interactivos que las transacciones de entrada se lleven a cabo sobre múltiples entradas u operaciones?			X				2
8.- ¿Se actualizan los archivos maestros de forma interactiva?						X	5
9.- ¿Son complejas las entradas, salidas, los archivos o las peticiones?				X			3
10.- ¿Es complejo el procesamiento interno?				X			3
11.- ¿Se ha diseñado código para ser reutilizado?						X	5

12.- ¿Están incluidas en el diseño la conversión y la instalación?	X	3
13.- ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	X	0
14.- ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizados por el usuario?	X	5
TOTAL	$\Sigma fi =$	48

Fuente: *Elaboración propia*

Para el ajuste se utiliza la ecuación:

$$PF = \text{Cuenta Total} * [\text{Grado de confiabilidad} + (\text{Tasas de Error} * \Sigma(fi))]$$

$$PF = \text{Cuenta Total} * [0.65 + (0.01 * \Sigma(fi))]$$

$$\text{Punto Función (PF)} = 296 * [0.65 + (0.01 * 48)]$$

$$\text{Punto Función (PF)} = 334.48$$

Para hallar el punto función ideal al 100% de los factores sería 70:

$$\text{Punto Función Max (PF)} = 296 * [0.65 + (0.01 * 70)]$$

$$\text{Punto Función Max (PF)} = 399.6$$

Calculando del % de funcionalidad Max:

$$\text{Funcionalidad} = [\text{Punto Función (PF)} / \text{Punto Función Max (PF)}] * 100\%$$

$$\text{Funcionalidad} = (334.48 / 399.6) * 100\%$$

$$\text{Funcionalidad} = 83.7 \%$$

Por tanto:

$$\text{FUNCIONALIDAD} = 84\%$$

Interpretando, la aplicación tiene una funcionalidad o utilidad del 84% para la empresa, lo que indica que él es sistema cumple con los requisitos funcionales de forma satisfactoria.

3.6.3 Confiabilidad

La confiabilidad permite evaluar la relación entre el nivel de funcionalidad y la cantidad de recursos usados, es decir, representa el tiempo que el software está disponible para su uso, la misma se calcula utilizando la privacidad de que un sistema presente fallas:

- **Comportamiento con respecto al tiempo:** Atributos de software relativos a los tiempos de respuesta y de procesamiento de los datos.
- **Comportamiento con respecto a Recursos:** Atributos software relativo a la cantidad de recursos usados y la duración de su uso en la realización de funciones.

La función a continuación muestra el nivel de confiabilidad del sistema:

$$F(t) = (\text{Funcionalidad}) * (e^{-\lambda * t})$$

Se ve el trabajo hasta que se observa un fallo en un instante t , la función es la siguiente:

$$\text{Probabilidad de hallar una falla: } (T \leq t) = (t)$$

$$\text{Probabilidad de no hallar una falla: } (T > t) = 1 - (t)$$

Donde:

$$\text{Funcionabilidad} = 84\% \rightarrow 0.8370$$

$$\lambda = 0.14 \text{ (es decir hay 1 error en cada 7 ejecuciones)}$$

$$t = 12$$

Hallamos la confiabilidad del sistema:

$$F(t) = (\text{Funcionalidad}) * (e^{-\lambda * t})$$

$$F(t) = 0.8370 * e^{-0.14 * 12}$$

$$F(t) = 0.8370 * e^{-1.68}$$

$$F(t) = 0.1559$$

Remplazando en la fórmula de no hallar una falla se tiene:

Probabilidad de fallos

$$P(T \leq t) = F(t)$$

$$(T \leq t) = 0.1559$$

$$\text{Probabilidad de fallos} = 0.1559 * 100$$

$$\text{Probabilidad de fallos} = 15.59\%$$

Probabilidad de éxito

$$(T > t) = 1 - (t)$$

$$(T > t) = 1 - 0.1559$$

$$(T > t) = 0.84$$

$$\text{Probabilidad de éxito} = 0.84 * 100$$

$$\text{Probabilidad de éxito} = 84\%$$

Siendo la probabilidad de fallo del 16% y la probabilidad de éxito de un 84%.

CONFIABILIDAD = 84%

3.6.4 Mantenibilidad

La Mantenibilidad se refiere a los atributos que permiten medir el esfuerzo necesario para realizar modificaciones al software, ya sea por la corrección de errores o por el incremento de funcionalidad. Para hallar mantenibilidad del sistema se utiliza el índice de madurez de software (IMS), que proporciona una indicación de la estabilidad de un producto de software.

Para calcular el índice hacen falta una serie de medidas anteriores:

Mt = número de módulos en la versión actual.(cada tabla)

Fc = número de módulos en la versión actual que han sido modificados.(1)

Fa = número de módulos en la versión actual que han sido añadidos.(0)

Fd = número de módulos de la versión anterior que se han eliminado en la versión actual.(1)

A partir de estas, el IMS (índice de madurez de software) se calcula de la siguiente forma:

Donde:

Mt = 10

Fc = 1

Fa = 0

Fd = 1

Entonces reemplazando en:

$$\mathbf{IMS = \frac{[M_T - (F_c + F_a + F_d)]}{M_T}}$$

$$\mathbf{IMS = \frac{[10 - (1 + 0 + 1)]}{10}}$$

$$\mathbf{IMS = 0.8}$$

MANTENIBILIDAD = 80%

3.6.5 Portabilidad

En este caso, se refiere a la habilidad del software de ser transferido de un ambiente a otro, y considera los siguientes aspectos:

- **Adaptabilidad**

Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.

- **Facilidad de Instalación**

Es el esfuerzo necesario para instalar el software en un ambiente determinado.

- **Conformidad**

Permite evaluar si el software se adhiere a estándares o convenciones relativas a portabilidad.

- **Capacidad de reemplazo**

Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.

Para el cálculo de la portabilidad se tomó en cuenta la siguiente tabla que contiene las características que se mencionaron anteriormente.

Tabla 27.
Cuadro de Portabilidad

Nº	FACTOR DE PORTABILIDAD	VALOR %
1	¿El sistema puede ser transferido de un lugar a otro?	85
2	¿Se puede adaptar con facilidad?	95
3	¿El sistema es fácil de instalar?	90
4	¿El sistema es fácil de utilizar?	95
5	¿Es capaz de reemplazar a una aplicación similar?	80
	TOTAL	89

Fuente: *Elaboración propia*

El sistema fue desarrollado con PHP y sistemas operativos, y la base de datos MySQL, que se ejecuta en todos los servidores web, se le da una calificación del 89% de portabilidad.

PORTABILIDAD= 89%

3.6.6 Resultados

De acuerdo a los resultados obtenidos se puede establecer la calidad total del sistema en base a los parámetros medidos anteriormente. La calidad está directamente relacionada con el grado de satisfacción con el usuario que ingresa al sistema.

Tabla 28.
Cuadro de Resultados

CARACTERISTICAS	RESULTADOS (%)
USABILIDAD	87
FUNCIONABILIDAD	84
CONFIABILIDAD	84
MANTENIBILIDAD	80
PORTABILIDAD	89
RESULTADO TOTAL	84.8

Fuente: *Elaboración propia*

EVALUACION DEL SISTEMA = 85%

3.7 COSTOS

En el análisis de costos del presente trabajo se ha considerado lo siguiente:

- Se obtuvo fuentes de información en línea y física, por el costo distribuido en la información.
- Costo de infraestructura el uso de servidor de la Universidad, es despreciable porque cuenta con propios servicios.
- Costos de recursos humanos por hora en la aplicación del modelo teórico y el desarrollo mismo, calculando 4 horas laborales por día.
- Se utilizó las herramientas para su implementación el software, cuyo costo de licencia es gratuita además con programas y servicios.

El proyecto no se apoyó en terceros económicamente, solo la inversión es de tiempo tanto en el sistema y en el modelo teórico, los resultados del proyecto son una justa inversión.

Tabla 29.
Estimación de costos del Software

ACTIVIDADES	COSTO
INVESTIGACIÓN	700 \$
ANÁLISIS, MODELO Y DISEÑO	650 \$
DESARROLLO DEL SISTEMA	3000\$
TOTAL	4350 \$

Fuente: *Elaboración propia*

3.7.1 Análisis de costos por el método Cócono II

El modelo de estimación de costos COCOMO II (Modelo Constructivo de Costos). Este modelo ayuda a estimar es costo del software. Para hallar el costo de construcción del software utilizaremos el valor obtenido de PF real siendo esta 634 utilizando el factor de conversión a KLDC de la tabla de conversión de puntos de función, siendo este como FACTOR; LDC/PF.

Tabla 30.
Conversión de Puntos de Función a KLDC

LENGUAJE	NIVEL	FACTOR LDC/PF
C	2.5	128
ANSI-BASIC	5	6464
JAVA	6	53
PL/I	4	80
ANSI COBOL 74	3	107
VISUAL BASIC	7.00	46
ASP	9.00	36
PHP	11.00	29
VISUAL C++	9.50	34

Fuente: *T-Chambi, 2007*

$$\text{LDC} = \text{PF real} * \text{Factor LDC/PF}$$

$$\text{LDC} = 334.48 * 29$$

$$\text{LDC} = 9\,699.92$$

$$\text{Siendo KLDC} = 1000$$

$$\text{KLDC} = 9699.92/1000$$

$$\text{KLDC} = 9.699$$

El modelo de costos COCOMO II ayuda a estimar el esfuerzo, tiempo, persona y costos.

Esta ecuación toma en cuenta para los tres modelos:

$$\mathbf{E} = \mathbf{a}(\mathbf{KL})^{\mathbf{b}} * \mathbf{m}(\mathbf{X}), \quad \textit{persona - mes}$$

$$\mathbf{Tdev} = \mathbf{c}(\mathbf{E})^{\mathbf{d}}, \quad \textit{en meses}$$

$$\mathbf{P} = \frac{\mathbf{E}}{\mathbf{Tdev}}, \quad \textit{en personas}$$

Donde:

E = Esfuerzo aplicado en persona – mes

Tdev = Tiempo requerido por el proyecto en meses cronológico

P = Número de personas requeridas

a, b, c, y d = constantes definidos en una tabla

KLDC = Número de líneas de código en miles

3.7.2 Costo del software

Se ha considerado bajo las líneas de código KLDC con la siguiente tabla.

Tabla 31.
Constantes del COCOMO II

PROYECTO DE SOFTWARE	a	b	c	d
Orgánico	2.4	1.05	2.5	0.38
Semi-acoplado	3.0	1.12	2.5	0.35
Empotrado	3.6	1.20	2.5	0.32

Fuente: *Elaboración propia*

Por tanto, el sistema desarrollado está entre el semi-acoplado porque es de tamaño y complejidad intermedia.

Tabla 32.
Variables factor de ajustes del esfuerzo (FAE)

Atributos	Valor					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos de software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de hardware						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
Atributos del proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	

Fuente: *Elaboración propia*

$$\mathbf{FAE} = 1,00 * 1,08 * 0,85 * 1,00 * 1,00 * 1,00 * 1,07 * 0,86 * 0,91 * 0,86 * 0,90 * 0,95 * 0,91 * 0,91 * 1,08$$

$$\mathbf{FAE} = 0,50551$$

Remplazando coeficientes del COCOMO II, se tiene:

Calculando persona – mes:

$$E = a * KLDC^b * FAE$$

$$E = 3 * (9.699)^{1.12} * 0.50551$$

$$E = 19.31 \text{ (persona-mes)}$$

Calculando en meses:

$$Tdev = c * (E)^d$$

$$Tdev = 2.5 * (19.31)^{0.35}$$

$$Tdev = 7.04 \text{ (meses)}$$

Por tanto, el cálculo de número de personas (**NP**) requeridas en el sistema es de:

$$NP = E \text{ (personas-mes)} / Tdev \text{ (meses)}$$

$$NP = 19.31 / 7.04$$

$$NP = 2.74 \text{ personas}$$

$$NP = 3 \text{ personas}$$

Si el salario aproximado que percibe un programador en nuestro medio es de 300\$us por mes o en Bs. 2 100, entonces se tiene:

$$NP * PAGO * T = \text{Costo del sistema}$$

$$CS = 4 * 300 * 7.04$$

$$CS = 6\ 336 \text{ $us}$$

CAPITULO VI

PRUEBAS Y RESULTADOS

4.1 PRUEBAS

4.1.1 Prueba de caja blanca

En esta prueba de la caja blanca está orientada al cálculo de las regiones que deben ser consideradas importantes como partes independientes del sistema puesta a prueba, estableciendo cuales son las entradas para que se ejecuten cada una de las regiones, asegurándose de esta manera que cada región se ejecute una vez, se debe seguir de forma general.

- Diseño del sistema para la elaboración del grafo.

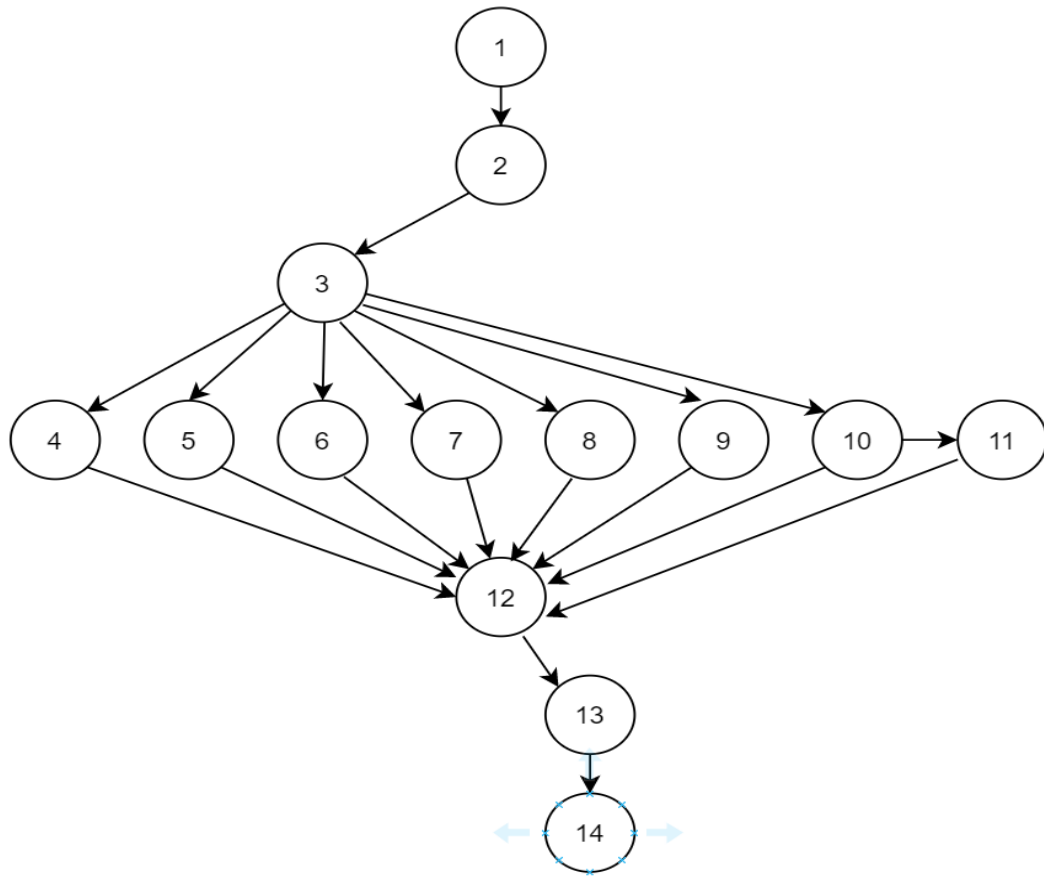


Figura 79. prueba de caja blanca

Fuente: *Elaboración propia*

Donde:

Inicio del programa (1)

Menú principal (2)

Módulo de registro (3)

Registro de usuario (4)

Registro de los viveros (5)

Registro de localización de los viveros (6)

Tipos de semillas (7)

Redes de agua (8)

Tipos de plagas (9)

Seguimientos (10)

Reporte (11)

Fin del ciclo de registro (12)

Fin del ciclo del sistema (13)

Fin del sistema (14)

- Una vez analizado el grafo generado, y a partir de ahí se procederá a determinar la complejidad ciclo mática del grafo mediante la formula siguiente.

Donde los valores de aristas y nodos son:

$$A = 20$$

$$N = 14$$

Reemplazando en $V(G) = A - N + 2$

$$V(G) = 20 - 14 + 2$$

$$V(G) = 8$$

- Una vez determinado las arista y nodos, determinamos el conjunto básico de los caminos detectados linealmente. Cada camino deberá ser probada con ciertas variables, los aminos encontrados son 8.

Determinando los caminos:

Camino 1: 1-2-3-4-12-13-14

Camino 2: 1-2-3-5-12-13-14

Camino 3: 1-2-3-6-12-13-14

Camino 4: 1-2-3-7-12-13-14

Camino 5: 1-2-3-8-12-13-14

Camino 6: 1-2-3-9-12-13-14

Camino 7: 1-2-3-10-12-13-14

Camino 8: 1-2-3-10-11-12-13-14

- Se debe prepara los casos de prueba para forzar la ejecución de cada uno de los caminos. Esta última condición establece que, que para la ejecución de estos, se deberán establecer las condiciones en la que al menos se ejecute los nodos establecidos de cada camino.

Tabla 33.
Tabla Camino 1

Descripción	El administrador (usuario) debe iniciar sesión con su cuenta y su contraseña para luego entrar al menú principal y registrar a los dueños de viveros.
Condición entrada	Si el usuario no está en la base de datos debe proceder con el registro de ellos como nuevo. Los campos que deben llenar son: Nombre, paterno, materno, c.i., dirección, teléfono, email, password, condición.
Resultado esperado	El sistema valida los datos de registrados.
Evaluación de la prueba	El sistema valido con éxito.

Fuente: *Elaboración propia*

Tabla 34.
Tabla Camino 2

Descripción	El administrador (usuario) está en el menú principal y registrar los detalles de los viveros.
Condición entrada	Si el dueño de los viveros no está en la base de datos se debe proceder con el registro de ellos como nuevo viveros. Los campos que deben llenar son: Nombre, encargado, capacidad, fase, tipo, superficie, dirección, latitud, longitud.
Resultado esperado	El sistema valida los datos de registrados.
Evaluación de la prueba	El sistema valido con éxito.

Fuente: *Elaboración propia*

Tabla 35.
Tabla Camino 3

Descripción	El administrador (usuario) está en el menú principal y registrar los detalles los de localización de los viveros.
Condición	Si la localización de los viveros no está en la base de datos se debe proceder con el registro de ellos como nuevo.
entrada	Los campos que deben llenar son: Nombre del departamento, nombre de la provincia y el municipio.
Resultado esperado	El sistema valida los datos de registrados.
Evaluación de la prueba	El sistema valido con éxito.

Fuente: *Elaboración propia*

Tabla 36.
Tabla Camino 4

Descripción	El administrador (usuario) está en el menú principal y debe registrar los detalles de tipos de semilla que se siembra en los viveros.
Condición	Si los viveros no cuentas con tipos de semillas registrado en la base de datos se debe proceder con el registro de ellos.
entrada	Los campos que deben llenar son: Nombre de la semilla, condición.
Resultado esperado	El sistema valida los datos de registrado
Evaluación de la prueba	El sistema valido con éxito.

Fuente: *Elaboración propia*

Tabla 37.
Tabla Camino 5

Descripción	El administrador (usuario) está en el menú principal y debe registrar los detalles de las
-------------	---

	redes de agua con las que cuenta el vivero sean estos naturales u otros.
Condición	Si los viveros no cuentan con un registro de redes de agua en la base de datos se debe proceder con el registro de ellos.
entrada	Los campos que deben llenar son: Marcar con que tipo de red cuenta
Resultado esperado	El sistema valida los datos de registrados.
Evaluación de la prueba	El sistema valido con éxito.

Fuente: *Elaboración propia*

Tabla 38.
Tabla Camino 6

Descripción	El administrador (usuario) está en el menú principal y debe registrar si los viveros tienen algún tipo de plaga existente en estos.
Condición	Si los viveros no cuentan con un registro de plagas existentes en la base de datos se debe proceder con el registro de estas plagas.
entrada	Los campos que deben llenar son: Nombre de la plaga, condición
Resultado esperado	El sistema valida los datos de registrados.
Evaluación de la prueba	El sistema valido con éxito.

Fuente: *Elaboración propia*

Tabla 39.
Tabla Camino 7

Descripción	El administrador (usuario) está en el menú principal, en esta parte del seguimiento se detalla la categoría de los empleados existentes en los viveros.
-------------	---

Condición	Si los viveros no cuentan con un registro de la categoría de sus empleados en la base de datos se debe proceder con el registro de ellos.
entrada	Los campos que deben llenar son: Ejecutor, contraparte, categoría, nativas, introducidas.
Resultado esperado	El sistema valida los datos de registrados.
Evaluación de la prueba	El sistema valido con éxito.

Fuente: *Elaboración propia*

Tabla 40.
Tabla Camino 8

Descripción	En esta parte el administrador (usuario) una vez llenado todos los campos que el sistema requiere, se puede imprimir un reporte con la información necesaria para los usuarios finales.
Condición	Para tener acceso al reporte de la información de los viveros se deben de llenar los campos que se piden.
Entrada	Los campos que deben llenar son:
Resultado esperado	El sistema valida todos los campos que se piden.
Evaluación de la prueba	El sistema realiza el reporte con éxito.

Fuente; *Elaboración propia*

4.1.2 Prueba de caja negra

En este punto de pruebas de caja negra o también conocidas como pruebas de comportamiento se centran en los requisitos funcionales del software. Para realizar la prueba de caja negra se tomara los casos de registro de viveros y Actualización de viveros.

Tabla 41.
Prueba de Caja Negra Registro De Viveros

Caso de prueba: registro de viveros	
Descripción	El registro se realiza primeramente verificando la información entregada por el dueño del vivero, para posteriormente registrar los datos del vivero y seleccionar su ubicación geográfica.
Condición de ejecución	El administrador deberá de estar autenticado dentro del sistema.
entrada	<ul style="list-style-type: none"> ➤ Elegir la opción en el menú de registro del sistema. ➤ Llenar el formulario de registro del vivero con los datos requeridos del mismo incluyendo su localización. ➤ Seleccionar el listado de viveros presionado el botón de listado de viveros en el cual se desplegara una lista actualizada de ellos y la opción de registra nuevo. ➤ Seguidamente luego de llenar todos los datos requeridos se debe de presionar en el botón guardar.
Resultado esperado	Luego de haber realizado las operaciones descritas en las entradas, deberá demostrar en listas de viveros registrados.
Evaluación de la prueba	El sistema realiza el registro con éxito.

Fuente: *Elaboración propia*

Tabla 42.
Prueba de Caja Negra Actualización de viveros

Caso de prueba: Actualización de viveros	
Descripción	El administrador podrá verificar los datos disponibles (existentes) del vivero y actualizarlos con datos actuales si así lo desea el dueño del vivero
Condición de ejecución	El administrador deberá de estar autenticado dentro del sistema.
entrada	<ul style="list-style-type: none"> ➤ Debe elegir en el menú la opción de editar vivero. ➤ Se debe de seleccionar los campos los cuales se van a actualizar. ➤ Seguidamente luego de llenar todos los datos requeridos se debe de presionar en el botón guardar.
Resultado esperado	Luego de haber realizado las operaciones descritas en las entradas, deberá demostrar que el vivero se actualizo correctamente.
Evaluación de la prueba	El sistema realiza la actualización con éxito.

Fuente: *Elaboración propia*

4.2 RESULTADOS

Ya finalizado con la pruebas realizadas al sistema para la “ABT”, se concluye que el sistema presento un comportamiento normal ante los eventos de inicio y salida de sesión de usuarios en la web.

En cuanto a la utilización del sistema, se pudo evidenciar la facilidad de la utilización por parte del personal de la institución, (administrador, usuario solicitantes del registro)

Realizando una comparación del antes y después a la implementación del sistema tendremos:

IMPLEMENTACION	
ANTES	DESPUES
Verificación de la información de los viveros al momento del registro.	La persona encargada de decepcionar el documento verifica los datos de localización del vivero.
La vista de los viveros es solo con la base de datos.	La vista de los viveros es más eficiente y rápida ya que se puede ver geográficamente (satelital).
No se contaba con la cantidad de producción de cada vivero	La cantidad de producción es real ya que esta se actualiza con frecuencia.
No se tenía un control del tipo de plantación de los viveros	Cuenta con un registro de tipo de semillas para un control de sus plantaciones.
No se contaba con la información de redes de agua	Se tiene un registro de redes de sgua para tener la información de riego de los viveros sean estos permanentes o temporales.

CAPITULO V

CONCLUSIONES Y RECOMENTACIONES

5.1 CONCLUSIONES

Una vez finalizado el proyecto de grado, en su desarrollo e implementación del **SISTEMA DE REGISTRO Y GEOLOCALIZACION DE VIVEROS EN EL DEPARTAMENTO DE LA PAZ, CASO: AUTORIDAD DE FISCALIZACIÓN Y CONTROL SOCIAL DE BOSQUES Y TIERRAS**. Para la institución “ABT”, se ha logrado alcanzar el objetivo principal planteado, bajo los requerimientos asignados. Tomando en cuenta los objetivos planteados se llega a las siguientes conclusiones:

En el objetivo principal: Desarrollar un Sistema de Registro Geolocalización de Viveros, que brinde una vista general y espacial en tiempo real de los existentes, tipos de especies, la cantidad de plantines, optimizando el procesamiento de la información en la ABT del departamento de La Paz para la toma de decisiones de forma oportuna.

- Se logró desarrollar el sistema para la ABT del departamento de La Paz para la toma de decisiones muy oportunas.

La aplicación agiliza los tiempos de acceso a la información.

En los objetivos secundarios:

- Se realizó el análisis de la institución.
- Se llegó a sistematizar la recolección de información referente a los viveros.
- Se logró a tener una vista general de todos los viveros registrados.
- Se tiene una vista espacial georreferenciada, la ubicación de los viveros existentes.
- Se llega a centralizar la información respecto a los viveros.
- Se diseñó la base de datos para almacenar toda la información referente a los viveros.
- Se optimizó el tiempo empleado para el registro de dueños de viveros.
- Se hizo más fiable el manejo de información sobre las características del vivero.
- Se realizó las pruebas pertinentes, como ser las pruebas funcionamiento.

Si mismo los reportes pueden ser exportados, a impresos y guardados en formato PDF o Excel.

De esta forma, se alcanzó el objetivo general y lograr la optimización de los procesos, la información ahora se encuentra a disposición de la institución para hacer el control adecuado a dichos procesos.

5.2 RECOMENDACIONES

A partir de este trabajo se propone las siguientes recomendaciones, con el fin de buscar el mejoramiento del sistema:

- Se recomienda con respecto al análisis y diseño del sistema, cuando se requiera la ampliación y creación de nuevos módulos, se recomienda primero revisar la documentación para poder tomar una buena decisión, ya que el sistema presenta elementos reutilizables que podrían ser utilizados en los módulos nuevos.
- Se recomienda para trabajos futuros con características similares utilizar algún framework, que facilite el desarrollo del producto.
- Se recomienda a la institución, implementar, utilizar y administrar el sistema de acuerdo a las instrucciones brindadas.
- La revisión periódica por cierto periodo de tiempo es recomendable para y un funcionamiento adecuado del sistema.

BIBLIOGRAFIA

- acensTechnologies. (2016). *Bootstrap, un framework para diseñar*. ACENS.
- Alzate, A. T. (2012). Obtenido de TEORÍA GENERAL DE SISTEMAS:
<http://bdigital.unal.edu.co/57900/1/teoriageneraldesistemas.pdf>
- Apaza, W. P. (2015). Trabajo Dirigido: Establecimiento De Viveros Forestales En Tres Municipios En El Marco Del Proyecto “Desarrollo Forestal De La Cuenca Del Lago Titicaca” .
- Arias, M. A. (2017). España: IT Campus Academy.
- ARISTASUR. (23 de 12 de 2010). Obtenido de Sistema de Coordenadas Geograficas:
<https://www.aristasur.com/contenido/sistema-de-coordenadas-geograficas-longitud-y-latitud>
- Aristides Dasso, A. F. (2014). Obtenido de iNtroduccion a la Programacion:
<http://proguno.unsl.edu.ar/apunte.pdf>
- Barcelona. (2004). Obtenido de Sistema De Informacion, Tecnologias De La Informacion:
<https://www.tdx.cat/bitstream/handle/10803/7001/08Jmmc08de12.pdf?sequence=8&isAllowed=y>
- Cantarero, J. A. (10 de 2016). *TRABAJO MONOGRAFICO*). Obtenido de Sistema de información web para la gestión de relación entre:
<http://ribuni.uni.edu.ni/1835/1/90197.PDF>
- Charles, M. (2004). *Desarrollo de aplicaciones web*. Barcelona: Funadacion Para La Universiada Oberta De Catalunya 1° Edicion.
- Chiavenato, I. (11 de 1999). Obtenido de Administracion de Recursos Humanos:
[http://www.ucipfg.com/Repositorio/MAES/MAES-08/UNIDADES-APRENDIZAJE/Administracion%20de%20los%20recursos%20humanos\(%20lect%202\)%20CHIAVENATO.pdf](http://www.ucipfg.com/Repositorio/MAES/MAES-08/UNIDADES-APRENDIZAJE/Administracion%20de%20los%20recursos%20humanos(%20lect%202)%20CHIAVENATO.pdf)
- Coelho, F. (17 de mayo de 2019). *Metodologia*. Obtenido de
<https://www.significados.com/metodo/>
- Collell, J. (s.a.). *CSS3 y Jvascript Avanzado*. España: Creative Commons.
- Conceptodefinicion.de. (14 de 11 de 2019). Obtenido de Definición de Registro:
<https://conceptodefinicion.de/registro/>
- Dans, E. (8 de 5 de 2013). *ED*. Obtenido de MariaDB, el software libre y el lucro cesante:
<https://www.enriquedans.com/2013/05/mariadb-el-software-libre-y-el-lucro-cesante.html>
- Departamento de Ciencia, Enginyeria Nautiques. (2004). Obtenido de Sistemas de Información Tecnologias de la Información:
<https://www.tdx.cat/bitstream/handle/10803/7001/08Jmmc08de12.pdf?sequence=8&isAllowed=y>
- Díaz, M. A. (2019). *Bases De Datos*. Obtenido de
<https://www.aiu.edu/cursos/base%20de%20datos/pdf%20leccion%201/lecci%C3%B3n%201.pdf>
- Eduard, H. (23 de 10 de 2012). Obtenido de METODOLOGÍA UWE-UML:
<https://elproyectedehernandezeduard.blogspot.com/2012/10/metodologia-uwe.html>
- Eichholz, M. (07 de 2017). Obtenido de Sistemas de Información Geográfica:
<https://www.bivica.org/files/informacion-geografica-guia.pdf>

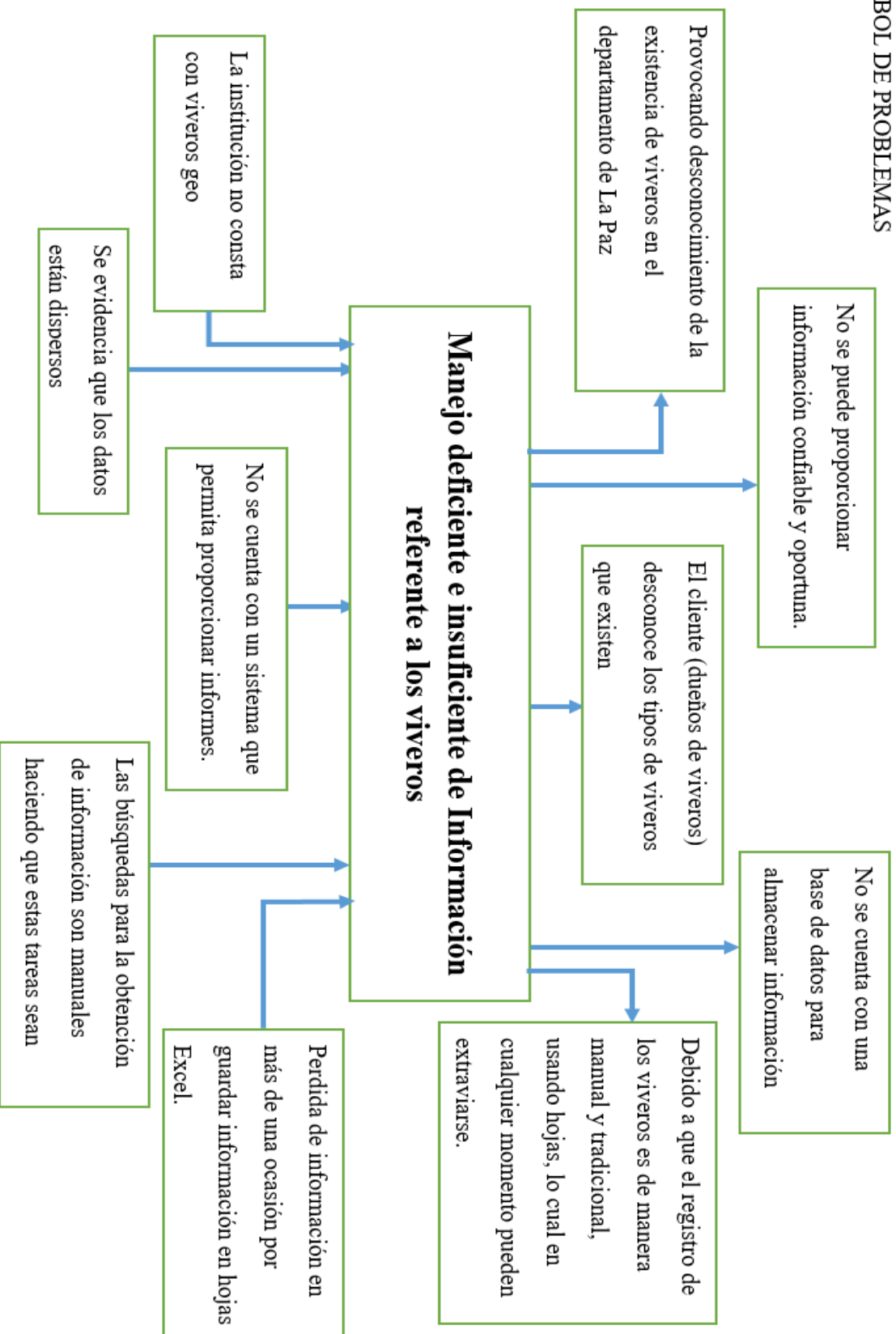
- Encada, C. (2018). *Analisis, desarrollo, implementacion eimplatacion de un sistema para la automatizacion de procesos en un centro medico*. Quito, Ecuador.
- Espinoza, B. R. (2017). Influencia De Un Sistema De geolocalización En el control Y Monitoreo de Vehículos con Dispositivos GPS En una Empresa Logística .
- Esquivel, V. M. (2013). Importancia de los Sistemas de Información Geográfica (SIG) en la Conservación .
- Francisco M. Gonzales, L. (2007). Obtenido de Introduccion a los Siste,mmas de Informacion, Fundamentos:
<https://www.uv.mx/personal/artulopez/files/2012/08/FundamentosSistemasInformacion.pdf>
- Fundacion, M. (2009). *MariaDB*. Obtenido de <https://mariadb.com/file/MariaDB.pdf>
- Gauchat, J. (2012). *El gran libro de HTML5, CSS y Javascript 1ªEdicion*. Barcelona: MARCOMBO.
- Gigch, V. (05 de 07 de 2016). Obtenido de Reingenieria de procesos:
<https://alondraalvarz.blogspot.com/2016/09/sistema-segun-autores-clasicos.html>
- González, E. \$. (2016). *Desarrollo de pagina Web en HTML 5 y Bootstrap*. Palploma.
- Gustavo, B. (2019). *Servidor Web*. Obtenido de <https://www.hostinger.es/tutoriales/quees-apache/>
- Huratdo, A. J. (2014). Obtenido de
<https://repositorio.umsa.bo/bitstream/handle/123456789/5605/T-2025.pdf?sequence=1&is>
- Hurtado, A. J. (2015). Tesis: Evaluacion De Seis Especies Forestales Bajo Tres Tratamientos Pregerminativos En Vivero Comunal, Sapecho – Alto Beni Umsa:.
- INSTITUTO NACIONAL TECNOLÓGICO. (2016). Obtenido de Manual del Protagonista Viveros y Semilleros:
https://www.academia.edu/36141451/VIVEROS_Y_SEMILLEROS_INSTITUTO_NACIONAL_TECNOL%3%93GICO_DIRECCI%3%93N_GENERAL_DE_FORMACI%3%93N_PROFESIONAL_MANUAL_DEL_PROTAGONISTA_ESPECIALIDAD_AGROPECUARIA_NIVEL_DE_FORMACI%3%93N_T%3%89CNICO_GENERAL
- Isau, A. (2015). *que es framework*. Obtenido de <https://openwebinars.net/blog/los-10-mejores-frameworks-php-que-solicitan-las-empresas/>
- Jorshua Alberto Aráuz Cantarero, 2016, TRABAJO MONOGRAFICO. (s.f.). “*Sistema de información web para la gestión de relación entre Plasencia Cigars y sus clientes*”.
- Koch Nora, A. K. (2001). en The Authoring Process Of The UML-Based Web Engineering Aproach . 1-29.
- Longatt, F. M. (2007). *Introduccion A Los Sistemas De Informacion*. Obtenido de <https://www.uv.mx/personal/artulopez/files/2012/08/FundamentosSistemasInformacion.pdf>
- Mamani, G. M. (2014). PROYECTO DE GRADO. “*Sistema Web De Control De Pedidos Y Ventas Caso: Empresa Itseven Soluciones Informáticas Integrales*”.
- Manual De Usuario Modulo Forestacion Y Reforestacion. (2020). “Sistema de monitoreo holístico e integral de bosques” Subsistema de Gestión y Desarrollo Forestal .
- Marcelo Arnold, P. y. (marzo de 1998). Obtenido de Introducción a los Conceptos Básicos de la Teoría General de Sistemas: <https://www.redalyc.org/pdf/101/10100306.pdf>
- Mendez, C. A. (2013). Tesis Efecto De Dos Tipos De Sustratos Y Tratamiento Pre-Germinacion Para Malva En El Vivero Forestal De Cota Cota Umsa.

- Minag. (2005). Plan Nacional De Reforestación. *Instituto Nacional De Reforestación*, Pag.2.
- Modelo de Calidad de Software para la DGSC.* (2011). Obtenido de <http://www.dgsc.go.cr>
- MUJERES, O. (31 de 10 de 2010). Obtenido de Principios Basicos de Programación Monitoreo y Evaluación: <https://www.endvawnow.org/es/articles/330-cul-es-el-monitoreo-y-la-evaluacin.html>
- Muñoz, E. F. (2007). Proyecto De Pre factibilidad Para La Creación De Un Vivero De Especies Nativas En El Municipio De Sesquile .
- Navall, M. (2016). Obtenido de El Vivero Forestal: <https://inta.gob.ar/sites/default/files/script-tmp-inta-viveroforestal.pdf>
- Núñez, M. J. (04 de 2012). Obtenido de GUÍA DE ESPECIES VEGETALES DEL VIVERO DE DOCENCIA Y EXTENSIÓN: http://www.saber.ula.ve/bitstream/handle/123456789/35273/tesis_maryn.pdf?sequence=1&isAllowed=y2012
- O'Brien, J. A. (2019). *Introduction To Information Systems In Business Management.* Obtenido de Introduction To Information Systems In Business Management <http://www.ucla.edu.ve/dcee/Departamentos/coordinaciones/informaticai/documentos/resumen%20tema1.pdf>
- Pila, J. P. (2013). Proyecto. *Automatización Del Seguimiento De Proyectos Y Contratos Realizados En El Instituto Superior De Investigaciones (Isi), En La Facultad De Ingeniería En Geología, Minas, Petroleos Y Ambiental .*
- Pressman. (2010). pag.609.
- Quiñones, J. R. (2015). *Manual Diseño Y Organizacion De Viveros.* Obtenido de <http://www.competitividad.org.do/wp-content/uploads/2016/05/Manual-de-Dise%C3%B1o-y-Organizaci%C3%B3n-de-Viveros.pdf>
- Reyes, J. (2015). *Manual De Organizacion De Viveros.* Obtenido de <http://www.competitividad.org.do/wp-content/uploads/2016/05/Manual-de-Dise%C3%B1o-y-Organizaci%C3%B3n-de-Viveros.pdf>
- Roger, P. R. (2010). Ingeniería De Spftware. En Pressman, *Ingenieria De Spftware* (pág. 609). Mexico: Mc Graw Hill.
- Sánchez, S. (2004). Desarrollo De Un Metodo ParaLa Calidad De Produccion De Plantulas En Viveros Forestales: Estudio De Casos Con Ciprés.
- Sarria, F. A. (2014). *Sistemas de Información Geográfica.* Obtenido de <http://fobos.inf.um.es/alonso/SIGCCAA/temario.pdf>
- Senn, J. A. (2019). *Sistemas De Informacion Para La Administracion.* Obtenido de <http://www.ucla.edu.ve/dcee/Departamentos/coordinaciones/informaticai/documentos/resumen%20tema1.pdf>
- Sierra, K. (2018). *que es laravel.* Obtenido de <https://www.synergyweb.es/blog/laravel-desarrollo-medida/>
- Tamayo, A. (2012). Obtenido de <http://bdigital.unal.edu.co/57900/1/teoriageneraldesistemas.pdf>
- Ucha, F. (06 de 2011). Obtenido de Definicion ABC Definición de Geolocalización: <https://www.definicionabc.com/geografia/geolocalizacion.php>
- Ucha, F. (09 de 2014). Obtenido de Definición de MONitoreo Ambiental: <https://www.definicionabc.com/medio-ambiente/monitoreo-ambiental.php>
- Ususario, M. d. (2019). Sistema de Monitoreo Holistico e Integral de Bosques. 2.
- Valdes, D. P. (2007). *javascript.* Obtenido de <http://www.maestrosdelweb.com/que-es-javascript/refiere>

Vaswani, V. (2010). *ACADEMIA*. Obtenido de Fundamentos de PHP:
https://www.academia.edu/10381049/Fundamentos_de_PHP

ANEXOS

ANEXO A
ARBOL DE PROBLEMAS



ANEXO B
ARBOL DE OBJETIVOS

