ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE LOS PROCESOS DE BOUQUETERÍA PARA UNA EMPRESA FLORÍCOLA DEL ECUADOR

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

CARLOS HUMBERTO LLUMIQUINGA HIDALGO
carlos_hllh@hotmail.com

DIRECTOR: ING. MSc. MYRIAM BEATRIZ HERNÁNDEZ ALVAREZ
myriam.hernandez@epn.edu.ec

Quito, Mayo 2015

DECLARACIÓN

Yo, Carlos Humberto Llumiquinga Hidalgo, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Carlos Humberto Llumiquinga Hidalgo

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Carlos Humberto Llumiquinga Hidalgo, bajo mi supervisión.

Ing. MSc. Myriam Hernández
DIRECTOR DE PROYECTO

AGRADECIMIENTOS

A la Poli, la Ing. MSc. Myriam Hernández por su guía y ayuda para el desarrollo de este proyecto.

A todas las personas que de una u otra forma colaboraron con el desarrollo del proyecto.

DEDICATORIA

A Dios por su guía, y darme fuerza para continuar.

A mis padres que son una guía y un apoyo incondicional en mi vida personal y profesional, a mis hermanas por su confianza, a mis sobrin@s por compartir su alegría conmigo.

A mis familiares y amig@s que siempre están al pendiente y alentándome.

CONTENIDO

CAPITULO 1. PLANTEAMIENTO DEL PROBLEMA 1

1.1 DESCRIPCIÓN DEL PROBLEMA 1
1.1.1 CONCEPTOS GENERALES 1
1.1.1.1 Bouquet 1
1.1.1.2 Flores de Línea 2
1.1.1.3 Flores Focales 2
1.1.1.4 Flores de Relleno 2
1.1.1.5 Flores Verdes 2
1.1.1.6 Accesorios 2

1.2 METODOLOGÍA DE DESARROLLO A APLICAR 3
1.2.1 PROCESO UNIFICADO DE DESARROLLO 3
1.2.1.1 Dirigido por Casos de Uso 3
1.2.1.2 Centrado en la arquitectura 3
1.2.1.3 Iterativo e Incremental 4
1.2.1.4 Enfocado en los riesgos 4
1.2.1.5 Ciclo de Vida del Proceso Unificado de Desarrollo 4
1.2.1.6 UML (Unified Modeling Language) 5
1.2.2 UWE (UML-Based Web Engineering) 5
1.2.2.1 Actividades de Modelado 6
1.2.2.1.1 Modelo de Requerimientos 6
1.2.2.1.2 Modelo de Contenido 6
1.2.2.1.3 Modelo de Navegación 7
1.2.2.1.4 Modelo de Presentación 8
1.2.2.1.5 Modelo de Procesos 9

1.3 DETERMINACION DE LAS HERRAMIENTAS A UTILIZAR 9
1.3.1 SOFTWARE LIBRE 9
1.3.1.1 Filosofía 10
1.3.2 GESTORES DE BASES DE DATOS 11
1.3.2.1 MySQL 11
1.3.2.1.1 Ventajas 11
1.3.2.1.2 Desventajas 12
1.3.2.2 PostgreSQL 12
1.3.2.2.1 Ventajas 12
1.3.2.2.2 Desventajas 12
1.3.2.3 Firebird 12
1.3.2.3.1 Ventajas 13
1.3.2.3.2 Desventajas 13
1.3.2.4 Ponderación 13
1.3.2.5 Selección del Gestor de Base de Datos 14
1.3.3 LENGUAJES DE PROGRAMACIÓN LIBRES 15
1.3.3.1 PHP 15
1.3.3.1.1 Ventajas 15
1.3.3.1.2 Desventajas 16
1.3.3.2 JSP 16
1.3.3.2.1 Ventajas 16
1.3.3.2.2 Desventajas 16

1.3.3.3 Python 16
1.3.3.3.1 Ventajas 17
1.3.3.3.2 Desventajas 17
1.3.3.4 Ponderación 17
1.3.3.5 Selección del lenguaje de programación 18
1.3.4 ENTORNO DE DESARROLLO INTEGRADO 18
1.3.4.1 Komodo Edit 19
1.3.4.1.1 Características 19
1.3.4.2 NetBeans PHP 19
1.3.4.2.1 Características 19
1.3.4.3 Aptana Studio 19
1.3.4.3.1 Características 20
1.3.4.4 Ponderación 20
1.3.4.5 Selección del Entorno de Desarrollo Integrado 20
1.3.5 HERRAMIENTAS CASE PARA UWE 21

CAPITULO 2. DESARROLLO DE LA APLICACIÓN 22

2.1 ANÁLISIS 22
2.1.1 ESPECIFICACIÓN DE REQUERIMIENTOS 22
2.1.1.1 Introducción 22
2.1.1.2 Alcance 22
2.1.1.3 Definiciones, acrónimos y abreviaciones 23
2.1.1.4 Referencias 23
2.1.1.5 Descripción General 23
2.1.1.5.1 Perspectiva de la Aplicación 23
2.1.1.5.2 Interfaces de Usuario 24
2.1.1.5.3 Hardware y Software 24
2.1.1.6 Características de los Usuarios 25
2.1.1.6.1 Perfiles de Usuario 25
2.1.1.7 Requerimientos Funcionales 25
2.1.1.7.1 Requerimiento Funcional #1 26
2.1.1.7.2 Requerimiento Funcional #2 26
2.1.1.7.3 Requerimiento Funcional #3 27
2.1.1.7.4 Requerimiento Funcional #4 28
2.1.1.7.5 Requerimiento Funcional #5 28
2.1.1.7.6 Requerimiento Funcional #6 29
2.1.1.7.7 Requerimiento Funcional #7 29
2.1.1.7.8 Requerimiento Funcional #9 30
2.1.1.7.9 Requerimiento Funcional #11 30
2.1.1.7.10 Requerimiento Funcional #12 31
2.1.1.7.11 Requerimiento Funcional #13 32
2.1.1.7.12 Requerimiento Funcional #14 32
2.1.1.7.13 Requerimiento Funcional #15 33
2.1.1.8 Exclusiones 33
2.1.1.9 Requerimientos no funcionales 33
2.1.2 MODELO DE REQUERIMIENTOS 34
2.1.2.1 Lista de Actores 34
2.1.2.2 Diagramas de Casos de Uso 35
2.1.2.2.1 Caso de uso Principal 35

2.1.2.2.2 Registrar Florícola ... *36*
2.1.2.2.3 Administrar Usuarios ... *36*
2.1.2.2.4 Administrar Áreas ... *36*
2.1.2.2.5 Administrar Clientes ... *37*
2.1.2.2.6 Administrar Pedidos De Bouquets ... *37*
2.1.2.2.7 Administrar Bouquets ... *38*
2.1.2.2.8 Administrar Accesorios ... *38*
2.1.2.2.9 Administrar Ocasión Bouquet ... *38*
2.1.2.2.10 Administrar Procesadoras ... *39*
2.1.2.2.11 Procesar Pedidos De Bouquets ... *39*
2.1.2.2.12 Administrar Flor ... *40*
2.1.2.3 Descripción de los Casos de Uso ... 40
2.1.2.3.1 Registrar Florícola ... *41*
2.1.2.3.2 Administrar Usuarios ... *42*
2.1.2.3.3 Administrar Áreas ... *43*
2.1.2.3.4 Administrar Clientes ... *45*
2.1.2.3.5 Administrar Pedidos De Bouquets ... *46*
2.1.2.3.6 Administrar Bouquets ... *48*
2.1.2.3.7 Administrar Accesorios ... *50*
2.1.2.3.8 Administrar Ocasión Bouquet ... *51*
2.1.2.3.9 Administrar Procesadoras ... *53*
2.1.2.3.10 Administrar Flor ... *59*
2.1.2.3.11 Administrar Tipo Flor ... *61*
2.1.2.3.12 Generar Reportes ... *62*
2.1.3 MODELO DE CONTENIDO ... 64
2.1.4 MODELO DE NAVEGACIÓN ... 65
2.1.4.1 Navegación Pedidos ... 66
2.1.4.2 Navegación Bouquets ... 67
2.1.4.3 Navegación Procesar Pedidos ... 68
2.1.4.4 Navegación Flor ... 69
2.1.4.5 Navegación Reportes ... 70
2.1.4.6 Navegación Administrar ... 71
2.1.5 MODELO DE PRESENTACION ... 72
2.1.5.1 Modelo de presentación Administrar ... 72
2.1.5.2 Modelo de presentación Pedidos Bouquets ... 73
2.1.5.2.1 Administrar ... *73*
2.1.5.3 Modelo de presentación Bouquets ... 74
2.1.5.3.1 Catálogo ... *74*
2.1.5.3.2 Administrar Bouquets ... *75*
2.1.5.4 Modelo de presentación Procesar Pedidos ... 76
2.1.5.4.1 Solicitar flor y Accesorios ... *76*
2.1.5.4.2 Asignar Elaboración ... *77*
2.1.5.4.3 Registrar Elaboración ... *78*
2.1.5.5 Modelo de presentación Administrar ... 79
2.1.5.5.1 Florícola ... *79*
2.1.5.6 Modelo de presentación Reportes ... 80
2.2 DISEÑO ... 80
2.2.1 ARQUITECTURA DE AKLLASISA ... 80
2.2.1.1 Arquitectura WEB ... 80

2.2.1.2 Patrón de diseño 81
2.2.1.3 Modelo de Datos 82
2.3 CONSTRUCCIÓN DE LA APLICACIÓN 84
2.3.1 PATRÓN DE DISEÑO MVC APLICADO A AKLLASISA 84
2.3.2 PROGRAMACIÓN MODULAR 84
2.3.3 ESTÁNDAR DE CODIFICACIÓN 85

CAPITULO 3. IMPLEMENTACIÓN Y PRUEBAS 86

3.1 IMPLEMENTACIÓN 86
3.1.1 HERRAMIENTAS UTILIZADAS 86
3.1.2 IMPLEMENTACIÓN DE AKLLASISA 86
3.1.3 CODIFICACIÓN 88
3.1.4 INSTALACIÓN DE AKLLASISA 97
3.1.4.1 Proceso de Instalación 97
3.1.4.2 Base de Datos 99
3.1.4.2.1 Datos Iniciales Tabla Área 99
3.1.4.2.2 Datos Iniciales Tabla Usuario 100
3.1.4.2.3 Datos Iniciales Tabla Tipo Flor 100
3.1.4.3 Archivos de Configuración de Akllasisa 100
3.1.4.4 Ingresar a la Aplicación 101
3.1.4.5 Datos de la Florícola 102
3.2 PRUEBAS 103
3.2.1 EVALUACIÓN CASO DE ESTUDIO 103
3.2.1.1 Caso de Estudio BellaFlor-Group 103
3.2.1.1.1 Caso de Prueba Administrar Usuarios 104
3.2.1.1.2 Caso de Prueba Administrar Clientes 107
3.2.1.1.3 Caso de Prueba Administrar Flor 110
3.2.1.1.4 Caso de Prueba Administrar Bouquets 113
3.2.1.1.5 Caso de Prueba Administrar Pedidos 117
3.3 EVALUACIÓN DE RESULTADOS 120
3.3.1 PRUEBAS DE ACEPTACIÓN 121

CAPITULO 4. CONCLUSIONES Y RECOMENDACIONES 126

4.1 CONCLUSIONES 126
4.2 RECOMENDACIONES 127

REFERENCIAS BIBLIOGRÁFICAS 128

ANEXOS DIGITALES 129

Anexo A 129
Anexo B 129
Anexo C 129
Anexo D 129
Anexo E 129

LISTA DE FIGURAS

Figura 1.1: Estereotipos UWE para el Modelo de Navegación 7
Figura 1.2**:** Estereotipos UWE para el Modelo de Presentación 9
Figura 1.3**:** Estereotipos UWE para el Modelo de Flujo de Procesos 9
Figura 2.1**:** Diagrama de CU Principal 35
Figura 2.2**:** Diagrama de CU Registrar Florícola 36
Figura 2.3**:** Diagrama de CU Administrar Usuarios 36
Figura 2.4**:** Diagrama de CU Administrar Áreas 36
Figura 2.5: Diagrama de CU Administrar Clientes 37
Figura 2.6**:** Diagrama de CU Administrar pedidos de bouquets 37
Figura 2.7**:** Diagrama CU Administrar Bouquets 38
Figura 2.8**:** Diagrama CU Administrar Accesorios 38
Figura 2.9: Diagrama CU Administrar Ocasión Bouquet 38
Figura 2.10: Diagrama de CU Administrar Procesadoras 39
Figura 2.11: Diagrama de CU Procesar Pedidos de Bouquets 39
Figura 2.12: Diagrama de CU Administrar Flor 40
Figura 2.13: Diagrama de Clases 64
Figura 2.14: Diagrama de Navegación Pedidos 66
Figura 2.15: Diagrama de Navegación Bouquets 67
Figura 2.16: Diagrama de Navegación Procesar Pedidos 68
Figura 2.17: Diagrama de Navegación Flor 69
Figura 2.18: Diagrama de Navegación Reportes 70
Figura 2.19: Diagrama de Navegación Administrar 71
Figura 2.20: Diagrama de presentación para Administrar 72
Figura 2.21: Diagrama de presentación para Pedidos Bouquets 73
Figura 2.22: Diagrama de presentación para Catálogo de Bouquets 74
Figura 2.23: Diagrama de presentación para Administrar Bouquets 75
Figura 2.24: Diagrama de presentación para Solicitar Flor y Accesorios 76
Figura 2.25: Diagrama de presentación para Asignar Elaboración 77
Figura 2.26: Diagrama de presentación para Registrar Elaboración 78
Figura 2.27: Diagrama de presentación Florícola 79
Figura 2.28: Diagrama de presentación Reportes 80
Figura 2.29: Arquitectura WEB 81
Figura 2.30: Diagrama de Secuencia MVC 82
Figura 2.31: Modelo Físico de datos de la aplicación Akllasisa 83
Figura 2.32: Patrón de diseño MVC aplicado a Akllasisa 84
Figura 3.1: Estructura de Programación modular aplicada a Akllasisa 87
Figura 3.2: Librería de Conexión a la Base de Datos 89
Figura 3.3: Modelos de la Tabla Flor 90
Figura 3.4: Modelo Flor.php 90
Figura 3.5: Modelo FlorDao.php 92
Figura 3.6: Controlador Módulo Flor 95
Figura 3.7: Controlador FlorController.php 95
Figura 3.8: Librerías 96
Figura 3.9: Terminal de Comandos 97
Figura 3.10: Ejecutar el Instalador 97
Figura 3.11: Usuario de Akllasisa 98
Figura 3.12: Extracción y permisos de Akllasisa 98
Figura 3.13: Ingresar IP del Servidor 98

Figura 3.14: Inicio XAMPP 99
Figura 3.15: Clave de root para MySQL 99
Figura 3.16: Archivo de configuración config.ini.php 101
Figura 3.17: Archivo de configuración db.ini 101
Figura 3.18: Página de Ingreso 102
Figura 3.19: Página Principal 102
Figura 3.20: Ingreso Datos de la Florícola 103
Figura 3.21: Resultados Pregunta 1 del Cuestionario 121
Figura 3.22: Resultados Pregunta 2 del Cuestionario 122
Figura 3.23: Resultados Pregunta 3 del Cuestionario 122
Figura 3.24: Resultados Pregunta 4 del Cuestionario 123
Figura 3.25: Resultados Pregunta 5 del Cuestionario 123
Figura 3.26: Resultados Pregunta 6 del Cuestionario 123
Figura 3.27: Resultados Pregunta 7 del Cuestionario 124
Figura 3.28: Resultados Pregunta 8 del Cuestionario 124
Figura 3.29: Resultados Pregunta 9 del Cuestionario 125
Figura 3.30: Resultados Pregunta 10 del Cuestionario 125

LISTA DE TABLAS

Tabla 1.1: Ciclo de Vida del Proceso Unificado 4
Tabla 1.2: Ponderación Gestores de Base de Datos 14
Tabla 1.3: Calificación Gestores de Base de Datos 15
Tabla 1.4: Ponderación Lenguajes de Programación Libres 18
Tabla 1.5: Calificación Lenguajes de Programación Libres 18
Tabla 1.6: Ponderación Entorno de Desarrollo Integrado 20
Tabla 1.7: Calificación Entorno de Desarrollo Integrado 20
Tabla 2.1: Lista de actores 34
Tabla 2.2: Descripción CU Registrar Florícola 41
Tabla 2.3: Descripción CU Administrar Usuarios 42
Tabla 2.4: Descripción CU Administrar Áreas 43
Tabla 2.5: Descripción CU Administrar Clientes 45
Tabla 2.6: Descripción CU Administrar Ordenes de Pedidos de Bouquets 46
Tabla 2.7: Descripción CU Administrar Bouquets 48
Tabla 2.8: Descripción CU Administrar Accesorios 50
Tabla 2.9: Descripción CU Administrar Ocasión Bouquet 52
Tabla 2.10: Descripción CU Administrar Procesadoras 53
Tabla 2.11: Descripción CU Solicitar Flor y Accesorios 55
Tabla 2.12: Descripción CU Asignar Elaboración de Bouquets 55
Tabla 2.13: Descripción CU Imprimir Etiquetas 57
Tabla 2.14: Descripción CU Asignar Flores y Accesorios 58
Tabla 2.15: Descripción CU Registrar Bouquet Elaborado 59
Tabla 2.16: Descripción CU Administrar Flor 60
Tabla 2.17: Descripción CU Administrar Tipo Flor 61
Tabla 2.18: Descripción CU Generar Reportes 63
Tabla 2.19: Capas de la Arquitectura WEB 81
Tabla 3.1: Herramientas Utilizadas 86
Tabla 3.2: Datos Iniciales Tabla Área 99
Tabla 3.3: Datos Iniciales Tabla Usuario 100
Tabla 3.4: Datos Iniciales Tabla Tipo Flor 100
Tabla 3.5: Prueba Ingresar Usuario 104
Tabla 3.6: Prueba Buscar Usuario 105
Tabla 3.7: Prueba Actualizar Usuario 105
Tabla 3.8: Prueba Eliminar Usuario 106
Tabla 3.9: Prueba Ingresar Cliente 107
Tabla 3.10: Prueba Buscar Cliente 108
Tabla 3.11: Prueba Editar Cliente 108
Tabla 3.12: Prueba Eliminar Cliente 109
Tabla 3.13: Prueba Ingresar Flor 110
Tabla 3.14: Prueba Buscar Flor 111
Tabla 3.15: Prueba Actualizar Flor 111
Tabla 3.16: Prueba Eliminar Flor 112
Tabla 3.17: Prueba Ingresar Bouquet 113
Tabla 3.18: Prueba Ingresar/Modificar Receta 114
Tabla 3.19: Prueba Buscar Bouquet 115
Tabla 3.20: Prueba Actualizar Bouquet 116
Tabla 3.21: Prueba Eliminar Bouquet 117
Tabla 3.22: Prueba Ingresar Pedido 117

Tabla 3.23: Prueba Buscar Pedido 118
Tabla 3.24: Prueba Actualizar Pedido 119
Tabla 3.25: Prueba Cancelar Pedido 120
Tabla 3.26: Muestra de Empleados que respondieron el Cuestionario 121

RESUMEN

Este proyecto detalla el proceso de desarrollo de Akllasisa (Flor Elegida), una aplicación web que permite administrar los procesos involucrados en la elaboración de un bouquet y las respectivas áreas de una florícola.

El desarrollo de la aplicación se realizó empleando el Proceso Unificado combinado con UWE mediante la elaboración de los modelos de navegación y presentación. Para la construcción de la aplicación se realizó una evaluación y selección de herramientas basadas en Software Libre.
Obteniendo como resultado una aplicación web que servirá de ayuda a una florícola para mejorar los procesos para la elaboración de un bouquet.

Akllasisa permite ingresar pedidos de bouquets, registrar recetas de los bouquets, solicitar la cantidad de flores y accesorios necesarios para elaborar bouquets, asignar y registrar la elaboración de bouquets, imprimir etiquetas de elaboración, registro de clientes, usuarios, procesadoras, flores, ocasión de un bouquet.
Además permite obtener reportes relacionados con la elaboración de bouquets en cada área involucrada.

PRESENTACIÓN

La aplicación web Akllasisa, se origina a partir de los procesos que interactúan en una florícola para elaborar un bouquet. Al no tener un medio de control y registro de los mismos ocasiona algunas veces el retraso o incluso no elaborar un pedido de los bouquets.
Akllasisa permite centralizar toda la información que se origina de los procesos de elaboración de un bouquet y poder usarla para beneficio de una florícola.
Para el fin y desarrollo del presente proyecto se lo ha dividido en 4 capítulos:

En el capítulo 1, se plantea el problema que presenta una Bouquetera respecto a la elaboración de un bouquet. Además de la justificación de la metodología y la selección de las herramientas basadas en software de libre difusión que se van a usar para el desarrollo de la aplicación.

En el capítulo 2, se encuentra el análisis, diseño y construcción de la aplicación. Conformados por: la especificación de requerimientos, diagramas y especificación de los casos de uso, diagramas de navegación, contenido y presentación, el diseño de la base de datos, codificación e interfaz que utilizarán los usuarios.

En el capítulo 3, se encuentra la implementación, pruebas y evaluación de resultados de Akllasisa. Tomando en cuenta los requerimientos de software y hardware ya establecidos para el correcto funcionamiento de la aplicación se procede a la implementación, pruebas y evaluación de los resultados.

En el capítulo 4, las conclusiones y recomendaciones que se obtuvieron a lo largo del desarrollo del proyecto.

CAPITULO 1. PLANTEAMIENTO DEL PROBLEMA

1.1 DESCRIPCIÓN DEL PROBLEMA

Muchas florícolas del país no cuentan con un sistema informático que permita controlar los procesos involucrados en la elaboración de los bouquets, generalmente la información relacionada es registrada manualmente o mediante hojas de cálculo. La interacción con esta información directa o indirectamente por medio de una aplicación puede mejorar el tiempo empleado en el proceso de elaboración de los bouquets.

En el caso de las recetas de Bouquets generalmente son almacenadas en documentos de texto u hojas de cálculo sin las respectivas políticas de respaldo originando la pérdida de dicha información.

Al no tener información detallada de los pedidos de elaboración de bouquets en determinada fecha o período, tal como: cuantos pedidos fueron cancelados, quien los elaboró, la cantidad de flor y accesorios empleados; dificulta tener una visión completa del proceso para poder analizar y realizar las respectivas correcciones y lograr obtener una mejor satisfacción del cliente.

1.1.1 CONCEPTOS GENERALES

1.1.1.1 Bouquet

Es una colección de flores en un arreglo creativo. Usados para decorar casas, edificios o como un obsequio para ocasiones y fechas especiales como cumpleaños, bodas, el día de la Madre, San Valentín, etc. Los arreglos de flores tienen una larga historia, la evidencia más antigua viene de Egipto, al igual que las pinturas de arreglos florales que se remontan al 2500 AC.

Cada flor en un arreglo o bouquet tiene su razón y propósito, las flores utilizadas para la creación de un arreglo floral se clasifican en:

1.1.1.2 Flores de Línea

Tienen perfiles altos y estrechos y se utilizan tanto vertical como horizontalmente para establecer la altura y la anchura de un arreglo. Como: gladiolos, delphiniums, fresias, snapdragons gigantes entre otras.

1.1.1.3 Flores Focales

Son las flores más grandes y llamativas en un arreglo. Como: Rosas, peonias, hortensias, crisantemos, lirios.

1.1.1.4 Flores de Relleno

Aquellas que poseen racimos de flores individuales en un solo tallo. Como: gysophilia (respiración de bebé), limonium, solidaster, proporcionan color y cuerpo entre las flores de línea y focales.

1.1.1.5 Flores Verdes

El verde agregado a un bouquet le da un aspecto más completo, rompe la monotonía de las flores, ofrece variados colores y texturas, agregan profundidad y enmarcan las flores. Dependiendo de la apariencia deseada, se agrega las hojas alrededor de los bordes de las flores o en el centro del arreglo se estructura el ramo, junto con las flores. Al seleccionar los verdes se debe escoger el que complementa el tamaño y la forma deseado del ramo.

Dentro de las Flores Verdes podemos mencionar: Eucalyptus, Bear Grass, Solidago, musgo.

1.1.1.6 Accesorios

Utilizados para completar la decoración o cambiar el estilo de un bouquet, los cuales se van agregando durante su elaboración y según sean requeridos de acuerdo a su diseño.

Los accesorios más comunes que se pueden usar son: papeles decorativos, brillantinas, bandas, lazos, recipientes, floreros, canastas, entre otros.

1.2 METODOLOGÍA DE DESARROLLO A APLICAR

Este trabajo tiene un objetivo académico y de investigación, presentando la necesidad de modificar los requisitos que aparezcan en el camino del desarrollo del mismo.

El Proceso Unificado de Desarrollo propuesto por Rumbaugh, Booch y Jacobson, es un modelo con un enfoque iterativo e incremental que se acopla satisfactoriamente para el desarrollo del presente trabajo.

1.2.1 PROCESO UNIFICADO DE DESARROLLO

Este proceso permite el desarrollo de un conjunto de actividades necesarias para transformar los requerimientos del usuario. Un marco de trabajo extensible que puede ser adaptado a organizaciones o proyectos específicos. Como sus características principales se pueden mencionar:

1.2.1.1 Dirigido por Casos de Uso

Un sistema de software se crea para cubrir las necesidades de los usuarios, conociendo de antemano que es lo que quieren y necesitan. “Usuario” no comprende solamente a los humanos sino la interacción con otros sistemas.
En el Proceso Unificado, los casos de uso se utilizan para capturar los requisitos funcionales y para definir los objetivos de las iteraciones. Guían el proceso de desarrollo (diseño, implementación y prueba). Los desarrolladores pueden crear una serie de modelos de diseño e implementación. Así los casos de uso se convierten en un medio que conduce el proceso de desarrollo.

1.2.1.2 Centrado en la arquitectura

Los casos de uso son desarrollados en conjunto con la arquitectura del sistema, de esta manera no se encargan en soledad del proceso de desarrollo. Cada uno recibe una influencia del otro.
La Arquitectura y los casos de uso están relacionados, los casos de uso deben encajar en la arquitectura, y a su vez la arquitectura debe ser lo suficientemente flexible para permitir el desarrollo de todos los casos de uso requeridos actualmente y a futuro. Ambos maduran a medida que el ciclo de vida continúa.

1.2.1.3 Iterativo e Incremental

La complejidad de un proyecto puede implicar mucho tiempo y esfuerzo. Dividirlo en partes más pequeñas o miniproyectos es mucho más práctico. Cada miniproyecto es una iteración y da como resultado un incremento. Los desarrolladores deciden que se va a implementar en una iteración, donde se tratarán los casos de uso que implican la utilidad del producto y tratar los riesgos importantes. Al ser un miniproyecto cada iteración realiza actividades específicas definidas en el ciclo de vida clásico: análisis, diseño, implementación y prueba.

1.2.1.4 Enfocado en los riesgos

Realizar continuos análisis de riesgos durante el desarrollo minimiza las posibilidades de fallos en cualquiera de las iteraciones o en el peor de los casos la cancelación del proyecto.

Obviamente los principales riesgos deben ser identificados en una etapa temprana del ciclo de vida del proyecto. Al tener los resultados de los análisis realizados en cada iteración se deben poner como una prioridad.

1.2.1.5 Ciclo de Vida del Proceso Unificado de Desarrollo

El proceso unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema, cada ciclo es una versión del sistema.

Cada ciclo consta de cuatro fases: inicio, elaboración, construcción y transición.

En la Tabla 1.1 se muestra el ciclo de vida del proceso unificado.

Tabla 1.1: Ciclo de Vida del Proceso Unificado

Inicio	Elaboración	Construcción	Transición

TIEMPO →

Inicio	Definir el alcance del proyecto
Elaboración	Planificar el proyecto, elaborar una arquitectura base, captura de requisitos
Construcción	Construir el sistema
Transición	Transición a los usuarios, correcciones

Elaborado por: Carlos Llumiquinga

1.2.1.6 UML (Unified Modeling Language)

Desarrollado a la par con el proceso unificado, surge por la necesidad de un lenguaje que ayude a comunicar ideas entre desarrolladores y también como apoyo en los procesos de análisis de un problema. UML se ha convertido en un estándar tan ansiado para representar y modelar información con la que se trabaja en las fases de análisis y especialmente de diseño.

1.2.2 UWE (UML-Based Web Engineering)

Aparece a finales de los 90s con la idea de buscar una manera estándar para la construcción de modelos para el análisis y diseño de sistemas web. Siendo el objetivo principal el utilizar un lenguaje común o definir asociaciones basadas en metamodelos (Rossi et al., 2008).

Propone una separación estricta de aspectos en las fases tempranas del desarrollo de la aplicación web e implementa un proceso de desarrollo basado en la construcción y transformación de modelos.

Los modelos son construidos en las diferentes fases del desarrollo: análisis, diseño e implementación, logrando interpretar diferentes vistas de la aplicación correspondientes a los diferentes aspectos (contenido, navegación y presentación). UWE propone al menos un tipo de diagrama UML para la visualización de cada modelo, para lograr representar las diferentes vistas de la aplicación web.

“Es una propuesta basada en UML y en el proceso unificado para modelar aplicaciones web. Está formada por una notación para especificar el dominio (basada en UML) y un modelo para llevar a cabo el desarrollo del proceso de modelado.” [1]

La característica distintiva de UWE es el cumplimiento de UML, ya que el modelo de elementos de UWE se define en términos de un perfil UML y como una extensión de meta-modelo de UML, por lo que permite el libre uso de cualquier diagrama UML que se conozca.

Así la notación de UWE es definida como un perfil de UML, que son una herramienta que permite extender UML para construir modelos de dominios particulares, mediante el uso de extensiones UML:

---

[1] Estudio de UWE, Universidad Carlos III de Madrid, 2009

Estereotipos: Permiten crear nuevos tipos de bloques de construcción que deriven de los existentes pero sean específicos a un problema representado entre corchetes angulares <<*NombreEstereotipo*>>.

Valores Etiquetados: Permiten agregar nueva información en la especificación de ese elemento, representado por (etiqueta, valor).

Restricciones: Permiten añadir nuevas reglas o modificar las existentes modificando el significado a un elemento de modelo usando el lenguaje OCL (Object Constraint Language) de UML o en lenguaje natural.

1.2.2.1 Actividades de Modelado

Las actividades base de modelado UWE son el análisis de requerimientos, modelo de requerimientos, el modelo de contenido, modelo de navegación y el modelo de presentación. A los cuales se pueden sumar otros modelos como el modelo de procesos.

1.2.2.1.1 Modelo de Requerimientos

Compuesto por los Caso de uso que se definen como cada interacción supuesta con el sistema a desarrollar, donde se representan los requisitos funcionales. En este modelo es donde se definirá que es lo que hace el sistema.
Los analistas utilizan diferentes tipos de técnicas de refinamiento para obtener una especificación más detallada de los requisitos funcionales, tales como flujos de trabajo, especificaciones formateadas, o prototipos. Estas representaciones suelen incluir actores, pre y postcondiciones, una descripción del flujo de trabajo, las excepciones y las situaciones de error, fuentes de información, los resultados de la muestra, y las referencias a otros documentos (Rossi et al., 2008).
De esta manera se pueden usar las descripciones detalladas de los casos de uso que incluyan los elementos mencionados anteriormente.

1.2.2.1.2 Modelo de Contenido

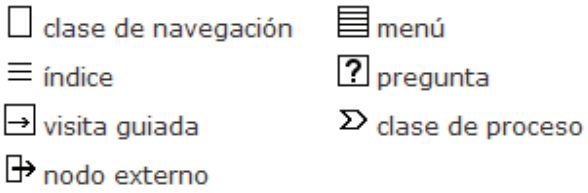
Provee una base para los Modelos de Diseño de una aplicación web, proporcionando un detalle visual de la información relevante del dominio.

"Especifica cómo se encuentran relacionados los contenidos del sistema, es decir, define la estructura de los datos que se encuentran alojados en el sitio web." [2]
El cual es representado por un diagrama UML de clases.

1.2.2.1.3 Modelo de Navegación

Por medio del análisis de requerimientos y el modelo de contenido, la estructura de navegación de la aplicación web es modelada. Las clases de navegación representan nodos navegables; los enlaces de navegación muestran enlaces directos entre las clases de navegación. Las rutas de navegación alternativas se manejan con menús. Primitivas de acceso se utilizan para llegar a múltiples instancias de una clase de navegación o seleccionar los elementos. En aplicaciones web que contienen lógica de negocio, los procesos de negocio deben ser integrados en la estructura de navegación. Los puntos de entrada y salida de los procesos de negocio son modelados por las clases de proceso en el modelo de navegación, el enlace entre ellas y con las clases de navegación se modela con enlaces de proceso. Cada clase de proceso está asociada a un caso de uso que modela un proceso del negocio (Rossi et al., 2008).
La Figura 1.1 muestra los estereotipos introducidos por UWE para este modelo:

Figura 1.1: Estereotipos UWE para el Modelo de Navegación



Fuente: http://uwe.pst.ifi.lmu.de

Las clases de contenido que se consideran relevantes para la navegación se seleccionan de entre el modelo de contenido, y estas clases, así como sus asociaciones se ponen en un modelo de navegación como clases de navegación y enlaces de navegación, respectivamente.
Los enlaces de navegación representan posibles pasos a seguir por el usuario, y por lo tanto estos enlaces deben ser dirigidos; si se desea la navegación hacia atrás

[2] Estudio de UWE, Universidad Carlos III de Madrid, 2009

y adelante entre las dos clases de navegación, una asociación se divide en dos. Los menús se añaden a cada clase de navegación que tiene más de una asociación saliente. Por último, las primitivas de acceso (índice, visitas guiadas y consultas) permiten la selección de una sola entidad de información, representada por una clase de navegación. Un índice, una visita guiada, o una consulta deben añadirse entre dos clases de navegación cada vez que la multiplicidad de la meta final de su asociación de enlace es mayor que uno. Las propiedades de la clase de contenido correspondientes a la clase de navegación sobre la que el índice o la consulta se ejecutan son agregadas como atributos de navegación a la clase de navegación. (Rossi et al., 2008)

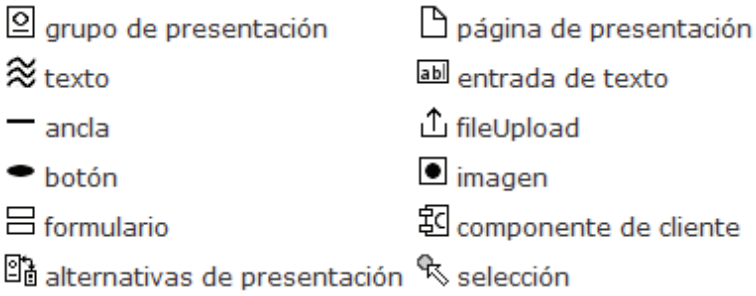
1.2.2.1.4 Modelo de Presentación

El modelo de presentación ofrece una visión abstracta de la interfaz de usuario (IU) de una aplicación Web. Basado en el modelo de navegación y resúmenes de aspectos concretos de la IU, como el uso de colores, fuentes, y la localización de los elementos de la IU en la página web; en su lugar el modelo de presentación describe la estructura básica de la IU, por ejemplo que elementos de la IU (texto, imágenes, anclas, formas) son usados para presentar los nodos de navegación.

Los elementos básicos de un modelo de presentación son las clases de presentación que se basan directamente en los nodos del modelo de navegación es decir, clases de navegación, menús, primitivas de acceso, y clases de proceso. Una clase de presentación está compuesta por elementos de la IU, como texto, anclas, botones, imágenes, formularios y colección de anclas (Rossi et al., 2008). Una página puede contener en su interior varios elementos de una clase de presentación y grupos de presentación.

En la Figura 1.2 se muestran los estereotipos que introduce UWE para este modelo:

Figura 1.2**:** Estereotipos UWE para el Modelo de Presentación



Fuente: http://uwe.pst.ifi.lmu.de

1.2.2.1.5 Modelo de Procesos

Se especificarán las tareas que realiza cada clase de proceso. Este modelo incluye:

- **Modelo de estructura de Procesos**: Describe las relaciones entre las diferentes clases de proceso.
- **Modelo de Flujo de Procesos:** Representado como un diagrama de actividades, donde se describe el comportamiento de una clase de proceso. UWE introduce los siguientes elementos:

En la Figura 1.3 se muestran los estereotipos que introduce UWE para este modelo:

Figura 1.3**:** Estereotipos UWE para el Modelo de Flujo de Procesos

Fuente: http://uwe.pst.ifi.lmu.de

1.3 DETERMINACION DE LAS HERRAMIENTAS A UTILIZAR

1.3.1 SOFTWARE LIBRE

“Richard Stallman, considerado el padre del Software Libre, autor de "Software libre para una sociedad libre", define a este tipo de software como aquel que una vez obtenido puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El cual suele estar disponible gratuitamente en Internet o a precio del coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y aunque conserve su carácter de libre, puede ser vendido comercialmente. Análogamente, al software gratuito denominado usualmente freeware (incluye en

algunas ocasiones el código fuente), este tipo de software no es libre en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa." [3]

1.3.1.1 Filosofía

De modo más preciso, el software libre se refiere a cuatro libertades que poseen sus usuarios:

- La libertad de usar el programa, con cualquier propósito.
- La libertad de estudiar cómo funciona el programa, y adaptarlo según necesidades. El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, promoviendo su uso.
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie.

Mucha gente reconoce el beneficio cualitativo del proceso de desarrollo de software cuando los desarrolladores pueden usar, modificar y redistribuir el código fuente de un programa. El movimiento del software libre hace especial énfasis en los aspectos morales o éticos del software, viendo la excelencia técnica como un producto secundario deseable de su estándar ético. El movimiento Open Source ve la excelencia técnica como el objetivo prioritario, siendo la compartición del código fuente un medio para dicho fin. Por dicho motivo, la Foundation Source Force (FSF) se distancia tanto del movimiento Open Source como del término "Open Source". Puesto que la OSI (Modelo de interconexión de sistemas abiertos) sólo aprueba las licencias que se ajustan a la OSD (Open Source Definition), la mayoría de la gente lo interpreta como un esquema de distribución e intercambia libremente "open source" con "software libre". Aun cuando existen importantes diferencias filosóficas entre ambos términos, especialmente en términos de las motivaciones para el desarrollo y el uso de tal software, inusualmente suelen tener impacto en el proceso de colaboración.

[3] Introducción al Software libre, Jesús González Barahona, 2003

Aunque el término "Open Source" elimina la ambigüedad de libertad frente a precio (en el caso del Inglés), introduce una nueva: entre los programas que se ajustan a la Open Source Definition, que dan a los usuarios la libertad de mejorarlos y los programas que simplemente tienen el código fuente disponible, posiblemente con fuertes restricciones sobre el uso de dicho código fuente. Con lo que se cree que cualquier software que tenga el código fuente disponible es open source, puesto que lo pueden manipular. Sin embargo, este tipo de software no da a sus usuarios la libertad de distribuir sus modificaciones, restringe el uso comercial, y en general restringe los derechos de los usuarios.

1.3.2 GESTORES DE BASES DE DATOS

Se realizará un análisis de tres gestores más representativos basados en software libre:

1.3.2.1 MySQL

Es un sistema de gestión de base de datos relacional, distribuido bajo licencia GPL de la GNU. Con un diseño multihilo que le permite soportar gran carga de forma muy eficiente. La empresa MySQL AB es su propietario, y mantiene el copyright del código fuente del servidor SQL, al igual que su marca.

MySQL AB también distribuye una versión de pago de MySQL cuya única diferencia radica en el soporte técnico que ofrece. Considerado como el gestor más usado en el mundo del software libre, por su gran rapidez y facilidad de uso. Debido a una infinidad de librerías y herramientas desarrolladas en muchos lenguajes de programación para facilitar su uso y administración.

1.3.2.1.1 Ventajas

- Su implementación multihilo, permite aprovechar la potencia de sistemas multiprocesador.
- Control de concurrencia.
- Amplio soporte para lenguaje SQL.
- Compatible con varios sistemas operativos (multiplataforma).

- Su conectividad, velocidad y seguridad hacen que sea la mejor opción para aplicaciones web.
- Su instalación y configuración no son complejas.
- Puede ser ejecutado en computadores con escasos recursos de hardware.

1.3.2.1.2 Desventajas

- Gran parte de sus utilidades no están documentadas.

1.3.2.2 PostgreSQL

Es un sistema de gestión de base de datos objeto-relacional basado en el proyecto POSTGRES, de la universidad de Berkley. El pionero de muchos conceptos existentes en el sistema objeto-relacional actual. Incluye características de la orientación a objetos, así como la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional.

Como muchos proyectos de software libre es dirigido por una comunidad de desarrolladores PGDG (PotsgreSQL Global Develoment Group).

1.3.2.2.1 Ventajas

- Soporta alta Concurrencia.
- Soporta transacciones (ACID) y claves foráneas (integridad referencial).
- Soporta triggers y procedimientos almacenados.
- Multiplataforma.
- Amplio soporte para lenguaje SQL

1.3.2.2.2 Desventajas

- Aumenta el consumo de memoria.
- Requiere de un cierto nivel de experiencia para su administración.
- La sintaxis de algunos de sus comandos o sentencias no es nada intuitiva.

1.3.2.3 Firebird

Es un sistema de administración de base de datos relacional (RDBMS) de código abierto, basado en la versión 6 de Interbase cuyo código fue liberado por Borland en el 2000. Dicho código fue reescrito de C a C++. El proyecto continua desarrollándose activamente.

1.3.2.3.1 Ventajas

- Realmente gratuito, puede ser utilizado sin ningún tipo de “honorarios” en aplicativos comerciales.
- Soporta transacciones ACID, integridad referencial.
- Bajos requerimientos de Hardware, bajo consumo de recursos.
- Multiplataforma.
- Control de concurrencia.

1.3.2.3.2 Desventajas

- Su administración, no se conocen muchas de las herramientas, la mayoría de ellas no presentan facilidad en su uso y otras son comerciales.
- Conectividad limitada en algunos lenguajes de programación.

1.3.2.4 Ponderación

Las siguientes características se tomarán como criterios para evaluar y seleccionar el gestor de base de datos para el desarrollo de la aplicación.
Si el gestor a evaluar cumple con la característica se asigna un puntaje de 1 y 0 si no cumple, excepto en las que se especifique el puntaje.

- Modelo de datos

  Relacional: Modelo utilizado para modelar problemas reales y administración dinámica de datos, tipo de datos no complejos.

  Objeto Relacional: Es el modelo relacional aplicado con los conceptos de objetos. Usado para el manejo de datos complejos como texto, imagen, mapas, audio y operaciones de mayor complejidad.

  Objetos: Incorpora todos los conceptos de orientación a objetos. Para el manejo de datos complejos como el caso de colecciones, tipos estructurados y objetos de gran tamaño.

  Para el presente estudio el gestor no necesita manipular tipos de datos complejos, por lo cual el puntaje se basará en este particular dando el mayor puntaje de 3 al modelo con menor complejidad en sus tipos de datos, 2 y 1 según aumente la complejidad.
- Multiplataforma

  El gestor puede funcionar sobre varios sistemas operativos.

- ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad) y claves foráneas.
  Garantiza que las transacciones en una base de datos se realicen de manera confiable.
- Control de concurrencia.
  Permite que varios usuarios puedan acceder a una misma base de datos al mismo tiempo.
- Lenguaje SQL
  Permite realizar definición, manipulación y control de las base de datos relacionales.
- Administración.
  Comprende el grado de dificultad para administrar el gestor de base de datos. Se asigna el puntaje de 3 si es fácil, 2 si presenta cierto grado de complejidad y 1 si es complejo.

En la Tabla 1.2, se muestra se muestra la ponderación a realizar.

Tabla 1.2: Ponderación Gestores de Base de Datos

Característica	Valor	Puntaje
Modelo de Datos	Relacional	3
	Objetos	1
	Objeto Relacional	2
Multiplataforma	Si	1
	No	0
ACID y claves foráneas	Si	1
	No	0
Control de Concurrencia	Si	1
	No	0
Administración	Fácil	3
	Normal	2
	Compleja	1

Elaborado por: Carlos Llumiquinga

1.3.2.5 Selección del Gestor de Base de Datos

De acuerdo a los criterios y los puntajes asignados en el punto anterior, se obtienen los siguientes resultados:

Tabla 1.3: Calificación Gestores de Base de Datos

Característica	MySQL	PosgreSQL	Firebird
Modelo de Datos	3	2	3
Multiplataforma	1	1	1
ACID y claves foráneas	1	1	1
Control de Concurrencia	1	1	1
Lenguaje SQL	1	1	1
Administración	3	1	2
TOTAL	**10**	**7**	**9**

Elaborado por: Carlos Llumiquinga

Los gestores muestran calificaciones casi parecidas diferenciándose en la administración de acuerdo a la complejidad que presentan. Para el presente trabajo se selecciona a MySQL de acuerdo a la evaluación realizada en la Tabla 1.3, además que se pretende dar a conocer herramientas basadas en software libre que representan una gran solución.

1.3.3 LENGUAJES DE PROGRAMACIÓN LIBRES

Existen diferentes lenguajes para programar aplicaciones web basadas en software libre, de acuerdo a las tendencias y necesidades de la aplicación a desarrollar. Las tecnologías siguen desarrollándose dando origen a nuevos problemas a dar solución. Dando origen a lenguajes de programación para web dinámicos, que permitan la interacción entre usuarios y con sistemas de bases de datos.
Para el presente trabajo se tomará en cuenta tres lenguajes, que son los más representativos para desarrollar aplicaciones web basadas en software libre.

1.3.3.1 PHP

Interpretado en el lado del servidor, permite la generación de páginas web dinámicas, en combinación con lenguaje HTML. Para su funcionamiento necesita de un servidor web como Apache.
La mayor parte de su sintaxis ha sido tomada del lenguaje C, Java y Perl con determinadas características.

1.3.3.1.1 Ventajas

- No necesita ser compilado.

- Fácil de aprender.
- Es un lenguaje muy rápido.
- Soporta orientación a objetos.
- Multiplataforma.
- Soporta conexión a MySQL, librerías nativas.
- Documentación completa desde su página oficial.

1.3.3.1.2 Desventajas

- Todo el trabajo lo realiza el servidor, resultando un poco ineficiente al presentarse muchas solicitudes.
- La comprensión del código se puede ver afectado al incluir sentencias de html dentro del código php.

1.3.3.2 JSP

Java Server Pages, desarrolla páginas web en Java. Multiplataforma, se ejecuta del lado del servidor desarrollado por Sun Microsystems. Permite la creación de aplicaciones web robustas. Su motor de páginas está basado en los servlets de Java. Necesita tener instalado un servidor Tomcat para su funcionamiento.

1.3.3.2.1 Ventajas

- Multiplataforma.
- Estructura en su código.
- Dispone de toda la potencia de java.

1.3.3.2.2 Desventajas

- Complejidad para aprenderlo si no se conoce java.
- Mucho mejor para proyectos grandes.
- Tiempos de desarrollo mayores que con otros lenguajes.
- En su primera ejecución necesita ser compilado.
- Soporta conexión a MySQL, mediante un conector.

1.3.3.3 Python

Los desarrolladores de software libre, lo consideran el lenguaje más limpio para programar. El código no necesita ser compilado, es interpretado. Los

programadores deben optar por un estilo de programación al usar python ya sea: programación orientada a objetos, estructurada, funcional u orientada a aspectos. Requiere de un servidor web como Apache con su modo python activo.

1.3.3.3.1 Ventajas

- Gran cantidad de funciones y librerías.
- El aprendizaje para programar aplicaciones web no es tan fácil.
- Multiplataforma.
- Orientado a objetos.

1.3.3.3.2 Desventajas

- Su configuración para servidores web es difícil.
- Si no existe un orden en su codificación se dificulta la lectura del código o tener errores indeseados.
- Necesita de un framework para hacer más sencillo su uso.
- Soporta conexión a MySQL, mediante un conector.

1.3.3.4 Ponderación

Las siguientes características se tomarán como criterios para evaluar y seleccionar el lenguaje de programación para el desarrollo de la aplicación.
Si el lenguaje a evaluar cumple con la característica se asigna un puntaje de 1, y 0 si no cumple, excepto en las que se especifique el puntaje.

- Dificultad de aprendizaje
  Que tan complicado es aprender a programar con el lenguaje. Se consideran 3 valores: bajo con un puntaje de 3, medio con un puntaje de 2 y alto con un puntaje de 1.
- Multiplataforma
  El lenguaje puede funcionar en cualquier sistema operativo.
- Orientado a Objetos
  El lenguaje implementa los conceptos establecidos para la programación orientada a objetos.
- Conexión con MySQL

Se asigna el puntaje de 2 al tipo de conexión nativa que sin necesidad de instalar y configurar un conector permita interactuar con MySQL y un puntaje de 1 si lo necesita.

Tabla 1.4: Ponderación Lenguajes de Programación Libres

Característica	Valor	Puntaje
Dificultad de aprendizaje	Bajo	3
	Medio	2
	Alto	1
Multiplataforma	Si	1
	No	0
Orientado a Objetos	Si	1
	No	0
Conexión con MySQL	Nativa	2
	Conector	1

Elaborado por: Carlos Llumiquinga

1.3.3.5 Selección del lenguaje de programación

De acuerdo a los criterios y los puntajes asignados en el punto anterior, se obtienen los siguientes resultados:

Tabla 1.5: Calificación Lenguajes de Programación Libres

Característica	PHP	JSP	Phyton
Dificultad de aprendizaje	3	2	1
Multiplataforma	1	1	1
Orientado a Objetos	1	1	1
Conexión a MySQL	2	1	1
TOTAL	**7**	**5**	**4**

Elaborado por: Carlos Llumiquinga

Para el presente trabajo se selecciona PHP, debido a los puntajes obtenidos en la Tabla 1.5.

1.3.4 ENTORNO DE DESARROLLO INTEGRADO

También llamado IDE por sus siglas en inglés (Integrated Development Environment), es un software compuesto por un conjunto de herramientas de programación. El cual puede dedicarse a un lenguaje de programación o varios.

Dentro de los entornos de desarrollo que poseen licencia de uso libre se analizarán los siguientes:

1.3.4.1 Komodo Edit

Responde a exigencias y búsqueda de herramientas para el desarrollo de web abierta. La filosofía de web abierta plantea la integridad de la web como un espacio público neutral independientemente de la plataforma, idioma y navegador.
Permite trabajar con distintos lenguajes de programación como PHP, Python, Perl, Ruby, C, C++, HTML, java, etc. Construido sobre el código base de Mozilla.

1.3.4.1.1 Características

- Resalta el código con distintos colores.
- Reconoce las diferencias entre sintaxis de cada lenguaje.
- Autocompletado de código.
- Multiplataforma.

1.3.4.2 NetBeans PHP

Escrito en Java es un producto libre y gratuito sin restricciones de uso. Permite crear aplicaciones Web con PHP 5, posee un potente debugger, va ganando terreno entre los desarrolladores LAMP.

1.3.4.2.1 Características

- Sincronización FTP.
- Chequeo de Sintaxis de código.
- Autocompletado y generación de Código.
- Resaltado de código por colores.
- Multiplataforma

1.3.4.3 Aptana Studio

Basado en Eclipse, permite trabajar con diferentes lenguajes de programación web como PHP, HTML, Java Script, CSS.

1.3.4.3.1 Características[4]

- Asistente de código para HTML y Javascript.
- Multiplataforma.
- Herramientas para trabajo con base de datos.
- Marcado de sintaxis mediante colores.
- Autocompletado de código.

1.3.4.4 Ponderación

El valor se asigna al cumplimiento o no de ciertas características del IDE, los puntajes se muestran en la Tabla 1.6.

Tabla 1.6: Ponderación Entorno de Desarrollo Integrado

Descripción	Puntaje
Si cumple	3
Mediante Plugins	2
Muy Limitado	1
No posee	0

Elaborado por: Carlos Llumiquinga

1.3.4.5 Selección del Entorno de Desarrollo Integrado

Tabla 1.7: Calificación Entorno de Desarrollo Integrado

Características	Komodo Edit	Netbeans PHP	Aptana Studio 3
Resalta Código	3	3	3
Sintaxis de Código	1	3	3
Autocompleta Código	1	3	3
Control de Versiones	2	3	2
Generación de Código	1	3	3
Multiplataforma	3	3	3
Depurador	1	3	2
TOTAL	**12**	**21**	**19**

Elaborado por: Carlos Llumiquinga

De acuerdo a los puntajes obtenidos en la Tabla 1.7 se selecciona Netbeans PHP.

[4] http://es.wikipedia.org/wiki/Aptana_Studio

1.3.5 HERRAMIENTAS CASE PARA UWE

Todas las herramientas CASE que soportan UML pueden ser usadas para la construcción de los modelos empleando los respectivos estereotipos. Existe una herramienta Open Source, UMLet 10.1 que ofrece los estereotipos usados en los modelos de navegación, presentación y procesos de UWE.

Para el modelamiento de Casos de Uso y Diagrama de Clases se usará la herramienta igualmente Open Source Modelio. Así se da a conocer más de una alternativa basada en Software libre para modelamiento.

CAPITULO 2. DESARROLLO DE LA APLICACIÓN

2.1 ANÁLISIS

2.1.1 ESPECIFICACIÓN DE REQUERIMIENTOS

2.1.1.1 Introducción

El presente documento contiene la información relacionada con el desarrollo de una aplicación web para facilitar los procesos involucrados en la elaboración de bouquets en una florícola, que han sido solicitados por sus clientes mediante una orden de pedido ingresada por el personal de ventas.

Se describirán y documentarán los diferentes procesos que se realizan para obtener un bouquet terminado y servirá en la utilización de la aplicación dentro de una florícola.

2.1.1.2 Alcance

La aplicación web a desarrollar, cubrirá los siguientes procesos detallados a continuación:

- Administración de la Aplicación.
  Permitirá al Administrador de la aplicación suministrar los datos necesarios para el funcionamiento de la aplicación e ingresar los datos de la florícola en la cual funcionará la aplicación.
- Administración de Usuarios.
  Facilitará al Administrador de la aplicación los procesos de: ingresar, modificar y eliminar usuarios de la aplicación.
- Administración de Clientes.
  Facilitará a los usuarios del departamento de ventas los procesos de: ingresar, modificar y eliminar los clientes que pertenecen a la florícola quienes realizan los pedidos de bouquets.
- Administración de Órdenes.
  Permitirá a los usuarios del departamento de ventas: ingresar, actualizar y cancelar de órdenes de bouquets solicitadas por los clientes.
- Administración de Bouquets.
  Permitirá al jefe de Bouquetera: administrar recetas y accesorios usados para la elaboración de bouquets, administrar las procesadoras que laboran

en el área Bouquetera, procesar los pedidos de los bouquets, solicitar la flor necesaria a la postcosecha, imprimir la programación, las etiquetas de elaboración y registrar los bouquets elaborados.

- Administración de Flor.
  Permitirá al Jefe de PostCosecha administrar la flor y tipos de flor disponibles en la florícola.
- Generación de Reportes.
  Permitirá la generación de reportes de acuerdo a parámetros ingresados y al tipo de usuario que lo necesite.

Utilizando la aplicación web se pretende obtener algunos beneficios enfocados a la automatización de procesos involucrados en la elaboración de los bouquets, optimización del tiempo y recursos empleados en dichos procesos.

2.1.1.3 Definiciones, acrónimos y abreviaciones

Se encuentran detalladas en el Anexo digital A.

2.1.1.4 Referencias

- Plantillas del proceso unificado de desarrollo.
- UWE, UML-Based Web Engineering.
- Recetas de Bouquets.
- Reglas del Negocio establecidas por la empresa, se encuentran en el Anexo digital B.

2.1.1.5 Descripción General

En este punto se describe las capacidades del sistema así como la integración a la red de la florícola.

2.1.1.5.1 Perspectiva de la Aplicación

La aplicación web a desarrollar brindará una automatización de los procesos relacionados con la elaboración y las órdenes de pedidos de bouquets. Integrando las áreas involucradas como son Bouquetera, Ventas y Gerencia.
Se podrá acceder a la aplicación mediante un navegador web como Mozilla Firefox, Google Chrome de preferencia y otros, por medio de la red LAN de la florícola.

2.1.1.5.2 Interfaces de Usuario

Las pantallas facilitarán al usuario navegar por medio de un menú vertical ubicado en la parte izquierda de la pantalla el cual será siempre visible.
Las pantallas contendrán secciones claramente definidas por medio de colores o fondos.
Dentro de las pantallas de administración se tendrá disponible las opciones básicas de ingresar, editar, eliminar y buscar ubicadas horizontalmente.

2.1.1.5.3 Hardware y Software

La aplicación para su funcionamiento necesitará de una red de computadoras, por medio de un esquema cliente servidor, con un computador actuando como servidor y las respectivas estaciones de trabajo de los usuarios.

a. Servidor

Debe cumplir con las siguientes características de hardware.

- Procesador Dual Core mínimo de 1.8 GHz.
- Memoria RAM mínima de 512 MB, recomendada de 1 GB o superior.
- Espacio en disco duro 20 GB.
- Monitor VGA.
- Tarjeta de red 10/100 Mbps o superior.

La aplicación para su correcto funcionamiento requerirá la instalación previa del siguiente software:

- Sistema operativo Linux (CentOS 5.7 o superior, Ubuntu 10.04 o superior en su versión LTS)
- PHP 5
- Servidor Web (Apache 2.0 o superior) con soporte para PHP5.
- DBMS MySQL.

Recomendable hacer uso de una solución conocida como XAMPP que contiene todo el software antes mencionado y otras utilidades integradas que facilitan la administración del entorno que necesita el servidor.

b. Cliente

Debe cumplir con las siguientes características de hardware:

- Procesador Pentium 4 mínimo de 1.8 GHz.
- Memoria RAM mínima de 512 MB, recomendada de 1 GB o superior.

- Espacio en disco duro 20 GB.
- Monitor VGA.
- Tarjeta de red 10/100 Mbps o superior.

La estación de trabajo deberá cumplir con los siguientes requisitos a nivel de software:

- Sistema Operativo basado en ventanas gráficas, Linux / Windows.
- Navegador web en versiones actuales, preferible Mozilla Firefox y Google Chrome (Windows), Google Chromium (Linux).

2.1.1.6 Características de los Usuarios

Los usuarios de la aplicación interactuará con las áreas de: ventas, bouquetera, postcosecha, gerencia y sistemas que pertenecen a una florícola. Realizarán las funciones de administración, pedidos, organización dentro de lo que a la elaboración de bouquets se refiere.

Los usuarios poseen una educación técnica y/o superior y poseen conocimientos básicos y avanzados en el uso de un computador e internet.

2.1.1.6.1 Perfiles de Usuario

A los usuarios que manejen la aplicación se los clasificará dentro de los siguientes perfiles:

- Administrador
- Ejecutivo de Ventas
- Jefe de Bouquetera
- PostCosecha
- Gerente

2.1.1.7 Requerimientos Funcionales

A continuación, se ofrece una descripción específica de la aplicación web, donde se podrá apreciar las entradas, salidas de los procesos involucrados, lo que brindará mayor información acerca de la aplicación.

2.1.1.7.1 Requerimiento Funcional #1

Registrar Florícola.

2.1.1.7.1.1 Introducción

El administrador de la aplicación ingresa los datos correspondientes a la florícola.

2.1.1.7.1.2 Entradas

- Razón Social de la florícola.
- RUC de la Florícola
- Dirección, teléfono, telefax, página web, dirección de correo electrónico.

2.1.1.7.1.3 Procesos

- Ingresar los datos solicitados de la florícola.
- Guardar los datos ingresados.

2.1.1.7.1.4 Salidas

- Confirmación del registro de los datos ingresados.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.2 Requerimiento Funcional #2

Administrar Usuarios

2.1.1.7.2.1 Introducción

El usuario administrador podrá agregar, modificar, borrar los usuarios y asignar un perfil dentro de la aplicación.

2.1.1.7.2.2 Entradas

- Nombre completo del usuario.
- Cédula de identidad del usuario.
- Nombre del usuario.
- Clave del usuario.
- Área de la florícola en que labora.
- Teléfono, domicilio, correo electrónico.

2.1.1.7.2.3 Procesos

- Registrar los datos ingresados del usuario en la aplicación.
- Registrar modificaciones de los datos de un usuario.
- Borrar un usuario de la base de datos.

2.1.1.7.2.4 Salidas

- Mensaje de confirmación del ingreso, modificación y eliminación del usuario, en la aplicación.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.3 Requerimiento Funcional #3

Administrar receta de bouquet.

2.1.1.7.3.1 Introducción

El jefe de la Bouquetera podrá agregar, modificar y eliminar un bouquet y su respectiva receta. Se registrará la receta genérica y los tipos de flores, accesorios y otros detalles que se emplearán para elaborarlo. Además de incluir una imagen digital del bouquet ya terminado.

2.1.1.7.3.2 Entradas

- Nombre del bouquet.
- Cantidad del Tipo de Flor
- Ocasión del bouquet.
- Accesorios.
- Observaciones.
- Imagen digital del bouquet terminado.
- Tipos de flor y Cantidad a usar (tallos).

2.1.1.7.3.3 Procesos

- Verificar que el nombre del bouquet no exista.
- Registrar datos del bouquet.
- Registrar Receta del Bouquet.

2.1.1.7.3.4 Salidas

- Confirmación del registro de los datos en la aplicación.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.4 Requerimiento Funcional #4

Administrar Accesorios.

2.1.1.7.4.1 Introducción

El jefe de la Bouquetera podrá agregar, modificar y eliminar los accesorios que serán utilizados para elaborar un bouquet.

2.1.1.7.4.2 Entradas

- Datos requeridos del accesorio.

2.1.1.7.4.3 Procesos

- Verificar que el accesorio no se haya registrado anteriormente.
- Registrar el accesorio en la aplicación.

2.1.1.7.4.4 Salidas

- Confirmación del ingreso de los datos del accesorio en la aplicación.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.5 Requerimiento Funcional #5

Administrar Ocasión Bouquet.

2.1.1.7.5.1 Introducción

El jefe de la Bouquetera podrá agregar, modificar y eliminar las ocasiones para las cuales se elaboran los bouquets.

2.1.1.7.5.2 Entradas

- Datos requeridos de la ocasión.

2.1.1.7.5.3 Procesos

- Verificar que la ocasión no se haya registrado anteriormente.
- Registrar la ocasión en la aplicación.

2.1.1.7.5.4 Salidas

- Confirmación del ingreso de la ocasión en la aplicación.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.6 Requerimiento Funcional #6

Registrar pedido de bouquets

2.1.1.7.6.1 Introducción

El Ejecutivo de Ventas registra en la aplicación los datos del pedido de bouquets.

2.1.1.7.6.2 Entradas

- Datos del cliente.
- Datos de la orden: modelo del bouquet, cantidad.
- Fecha de salida.

2.1.1.7.6.3 Procesos

- Seleccionar el cliente que hace el pedido de bouquets.
- Agregar el o los bouquets solicitados y la cantidad.
- Verificar que la fecha de salida del pedido cumpla las reglas del negocio (Anexo digital B).
- Guardar los datos del pedido.

2.1.1.7.6.4 Salidas

- Confirmación del registro de la orden en la aplicación.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.7 Requerimiento Funcional #7

Solicitar flor y accesorios.

2.1.1.7.7.1 Introducción

Permite al jefe de la Bouquetera realizar una solicitud de la cantidad de flor y un listado de los accesorios necesarios para cubrir los pedidos de bouquets para determinada fecha.

2.1.1.7.7.2 Entradas

- Fecha de salida de las órdenes.

2.1.1.7.7.3 Procesos

- Calcular la cantidad de flor y accesorios necesarios para elaborar las órdenes a la fecha seleccionada.

2.1.1.7.7.4 Salidas

- Listado con los tipos y cantidad de flor que se necesitan.
- Listado con los accesorios que se necesitan.

2.1.1.7.8 Requerimiento Funcional #9

Administrar Flor.

2.1.1.7.8.1 Introducción

Permite al jefe de la postcosecha agregar, modificar, borrar los tipos de flor disponibles en la florícola.

2.1.1.7.8.2 Entradas

- Información requerida de la flor.

2.1.1.7.8.3 Procesos

- Ingresar, modificar y eliminar los datos ingresados de la flor en la base de datos.

2.1.1.7.8.4 Salidas

- Confirmación del ingreso, modificación y eliminación de la flor, en la base de datos.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.9 Requerimiento Funcional #11

Asignar elaboración de bouquets.

2.1.1.7.9.1 Introducción

Permite al jefe de la Bouquetera distribuir la elaboración de los bouquets de las diferentes órdenes a las procesadoras, priorizando volúmenes, clientes, dificultad de los modelos, tiempo y recursos disponibles.

2.1.1.7.9.2 Entradas

- Bouquets a elaborar.
- Datos de la procesadora.
- Cantidad del bouquet a elaborar que se asignara a la procesadora.

2.1.1.7.9.3 Procesos

- Seleccionar los bouquets a elaborar.
- Asignar la cantidad y el bouquet a la procesadora para su elaboración.

2.1.1.7.9.4 Salidas

- Código de elaboración del bouquet.
- Impresión de etiqueta, la cual debe contener:
    - Código de Elaboración.
    - Nombre de la Florícola.
    - Nombre del Bouquet.
    - Razón Social del Cliente.
    - Dirección del Cliente.
- Reporte de programación a la fecha por procesadora.

2.1.1.7.10 Requerimiento Funcional #12

Registrar elaboración de bouquet.

2.1.1.7.10.1Introducción

Permite al jefe de la Bouquetera registrar el bouquet elaborado por una procesadora de acuerdo al código de elaboración.

2.1.1.7.10.2Entradas

- Datos de la procesadora.
- Código de Elaboración.

2.1.1.7.10.3Procesos

- Verificar datos del código de elaboración del bouquet.

2.1.1.7.10.4Salidas

- Confirmación del registro de elaboración del bouquet.

2.1.1.7.11 Requerimiento Funcional #13

Administrar Clientes

2.1.1.7.11.1Introducción

Permite al Ejecutivo de Ventas agregar, modificar, borrar los clientes de la florícola.

2.1.1.7.11.2Entradas

- Información requerida del cliente.

2.1.1.7.11.3Procesos

- Ingresar, modificar y eliminar los datos ingresados del cliente en la base de datos.

2.1.1.7.11.4Salidas

- Confirmación del ingreso, modificación y eliminación de los clientes, en la base de datos.
- De existir algún error la aplicación mostrará un mensaje.

2.1.1.7.12 Requerimiento Funcional #14

Administrar Procesadoras

2.1.1.7.12.1Introducción

Permite al jefe de Bouquetera agregar, modificar, borrar las procesadoras que tiene en su área.

2.1.1.7.12.2Entradas

- Información requerida de la procesadora.

2.1.1.7.12.3Procesos

- Ingresar, modificar y eliminar los datos ingresados de la procesadora en la base de datos.

2.1.1.7.12.4Salidas

- Confirmación del ingreso, modificación y eliminación de la procesadora en la base de datos.

2.1.1.7.13 Requerimiento Funcional #15

Generar Reportes

2.1.1.7.13.1Introducción

Permite a los usuarios obtener reportes relacionados con el proceso de elaboración de las órdenes de bouquets.

2.1.1.7.13.2Entradas

- Seleccionar el reporte.
- Ingresar parámetros para la elaboración del reporte.

2.1.1.7.13.3Procesos

- Seleccionar la información solicitada de acuerdo a los parámetros ingresados.

2.1.1.7.13.4Salidas

- Impresión en pantalla y física de la información obtenida.

2.1.1.8 Exclusiones

La aplicación a desarrollar muestra las funcionalidades esenciales que poseerá en torno al proceso de elaboración de bouquets y adoptará las siguientes exclusiones:

- No realizará el proceso de facturación de los bouquets.
- No detallará ningún tipo de valor económico respecto a las órdenes ingresadas para la elaboración de los bouquets.

2.1.1.9 Requerimientos no funcionales

A continuación se especifica la manera como debe ser implementado el sistema:

- La aplicación debe poseer una interfaz amigable e intuitiva para el usuario.
- Se debe poder acceder a la aplicación desde cualquier computador independiente del sistema operativo y navegador que posea desde la red interna de la florícola.

2.1.2 MODELO DE REQUERIMIENTOS

En UWE el modelo de requerimientos[5] consiste en:

- Casos de Uso, compuesto por la lista de actores y los diagramas de Casos de Uso.
- Descripción de los Casos de Uso por medio de Diagramas de actividad o una descripción muy detallada de los mismos.

2.1.2.1 Lista de Actores

En la Tabla 2.1 se muestran los actores relacionados con la aplicación, su respectiva descripción y actividades dentro de la misma.

Tabla 2.1: Lista de actores

No.	Actor	Descripción
1.	Administrador (de la Aplicación)	Encargado de la configuración y mantenimiento de los parámetros básicos para que funcione la aplicación por ejemplo: datos de la Florícola, usuarios.
2.	Ejecutivo de Ventas	Se encarga de ingresar las órdenes de pedidos de bouquets, administrar los clientes y generar reportes que necesite.
3.	Jefe de Bouquetera	Se encarga de ingresar las recetas de los bouquets, administrar los accesorios, procesadoras, solicitar la flor necesaria para elaborar los bouquets, impresión de etiquetas de elaboración, registrar los bouquets elaborados.

[5] Fuente: http://uwe.pst.ifi.lmu.de/teachingTutorialRequirements.html

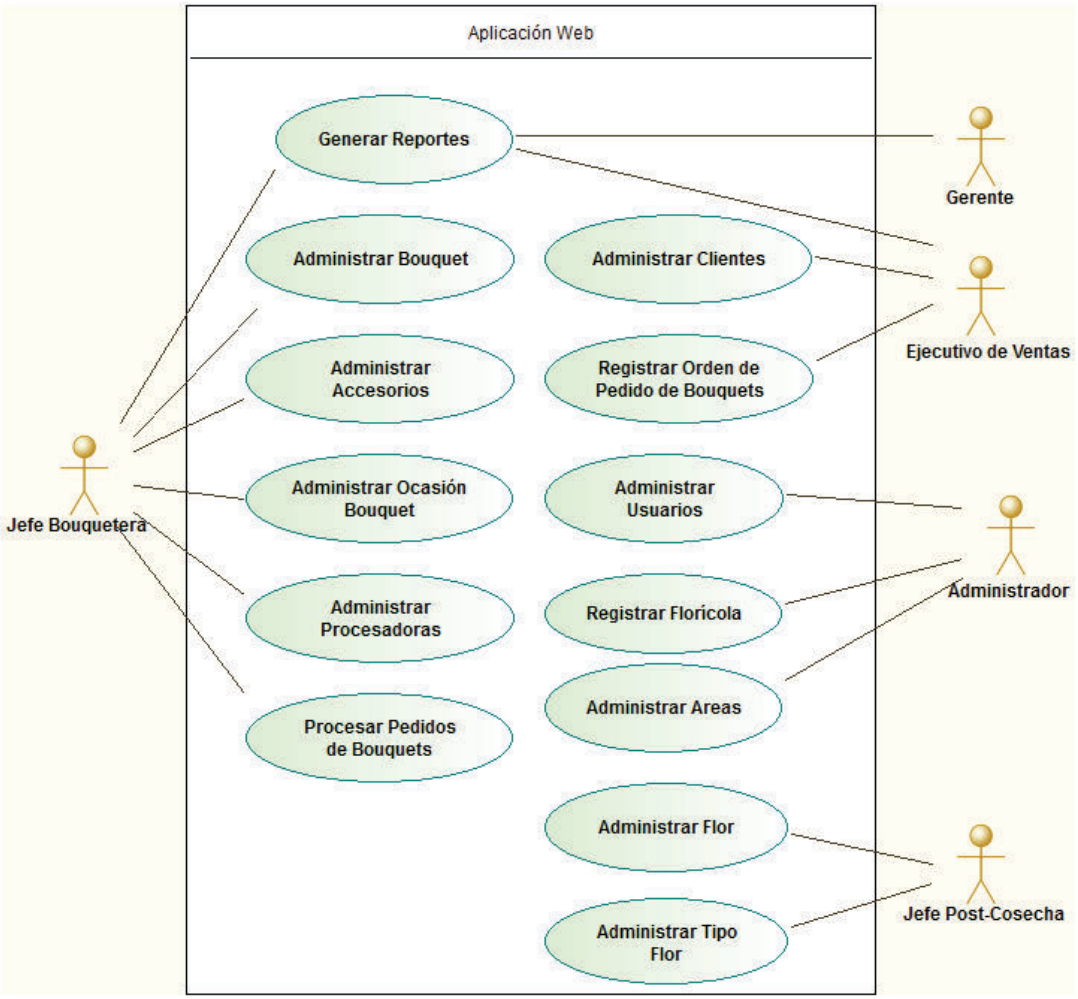
4.	Jefe de PostCosecha	Se encarga de administrar la flor y tipos de flor disponible en la florícola.
5.	Gerente	Se encarga de generar reportes relacionados con el proceso de elaboración de los bouquets

Elaborado por: Carlos Llumiquinga

2.1.2.2 Diagramas de Casos de Uso

2.1.2.2.1 Caso de uso Principal

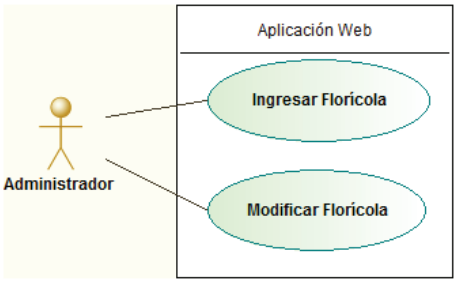
Figura 2.1**:** Diagrama de CU Principal



Elaborado por: Carlos Llumiquinga

2.1.2.2.2 Registrar Florícola

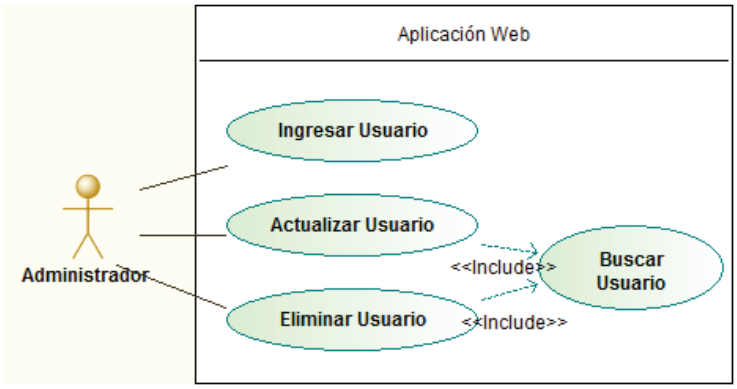
Figura 2.2: Diagrama de CU Registrar Florícola



Elaborado por: Carlos Llumiquinga

2.1.2.2.3 Administrar Usuarios

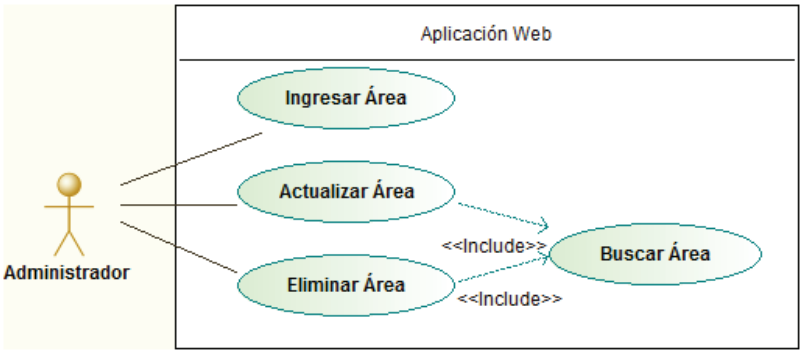
Figura 2.3: Diagrama de CU Administrar Usuarios



Elaborado por: Carlos Llumiquinga

2.1.2.2.4 Administrar Áreas

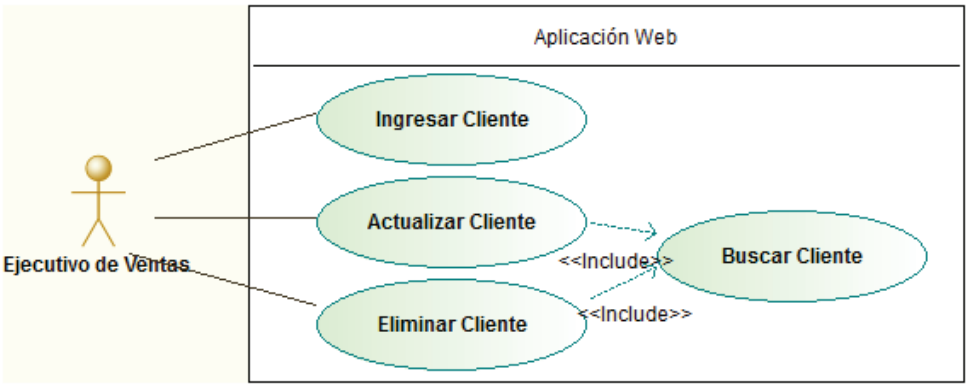
Figura 2.4: Diagrama de CU Administrar Áreas



Elaborado por: Carlos Llumiquinga

2.1.2.2.5 Administrar Clientes

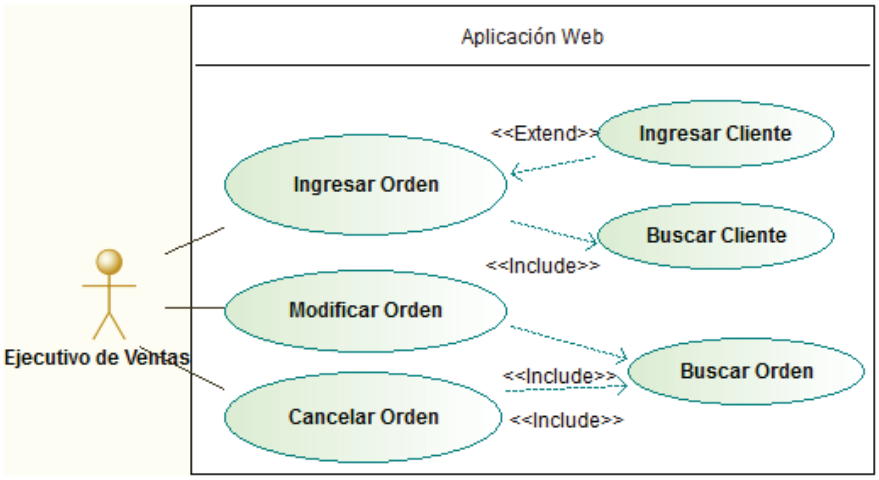
Figura 2.5: Diagrama de CU Administrar Clientes



Elaborado por: Carlos Llumiquinga

2.1.2.2.6 Administrar Pedidos De Bouquets

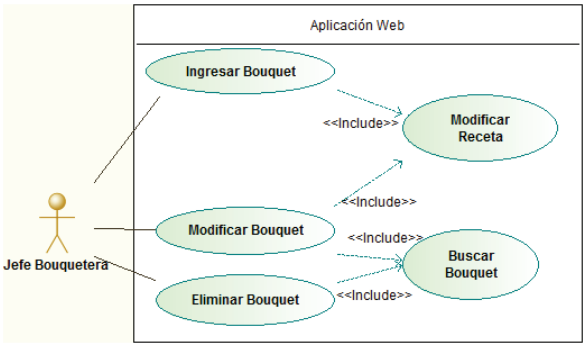
Figura 2.6**:** Diagrama de CU Administrar pedidos de bouquets



Elaborado por: Carlos Llumiquinga

2.1.2.2.7 Administrar Bouquets

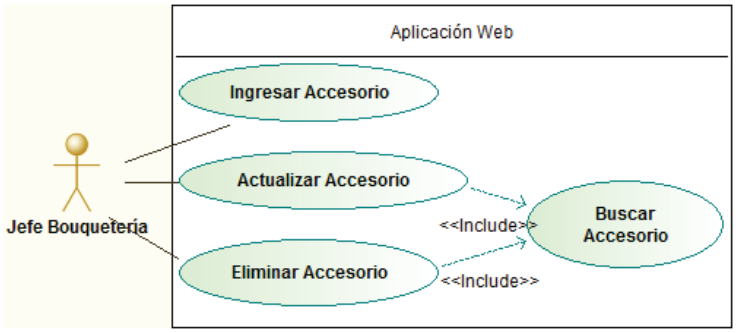
Figura 2.7**:** Diagrama CU Administrar Bouquets



Elaborado por: Carlos Llumiquinga

2.1.2.2.8 Administrar Accesorios

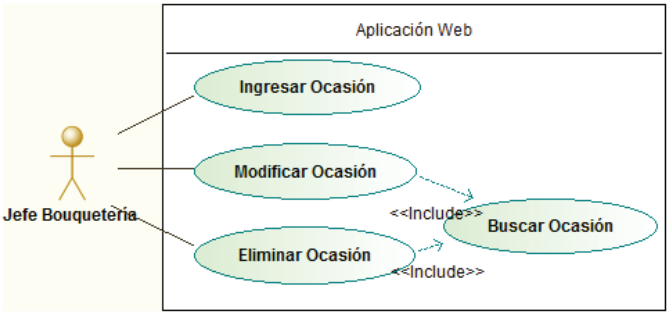
Figura 2.8**:** Diagrama CU Administrar Accesorios



Elaborado por: Carlos Llumiquinga

2.1.2.2.9 Administrar Ocasión Bouquet

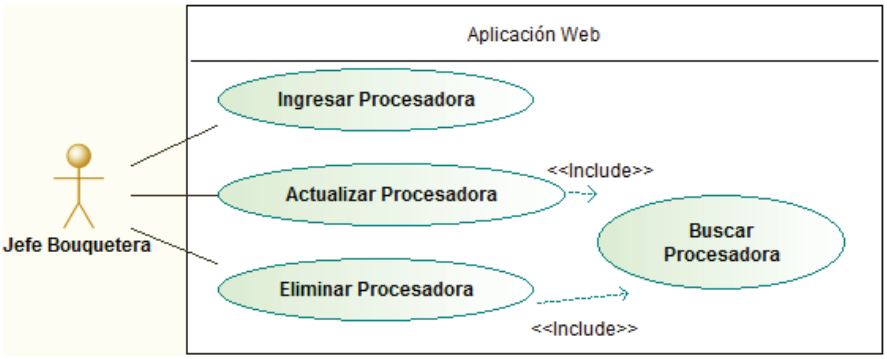
Figura 2.9: Diagrama CU Administrar Ocasión Bouquet



Elaborado por: Carlos Llumiquinga

2.1.2.2.10 Administrar Procesadoras

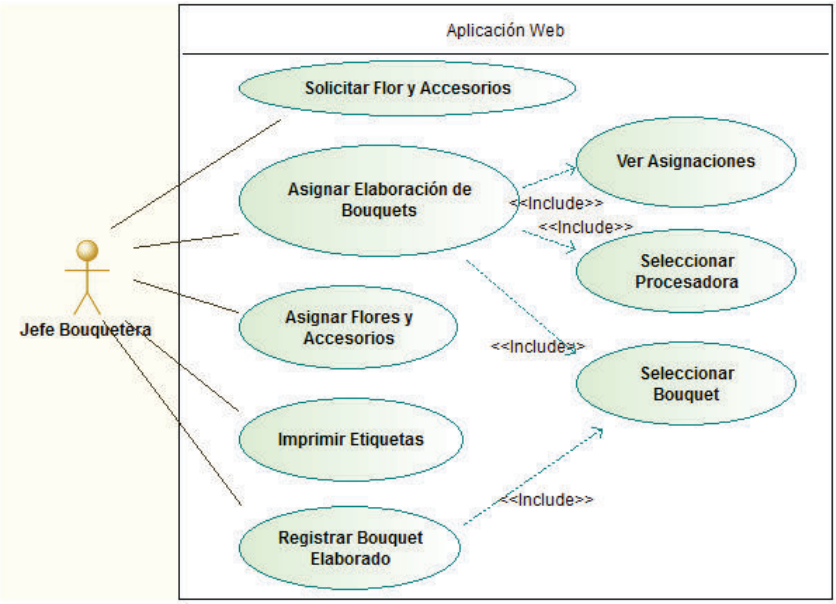
Figura 2.10: Diagrama de CU Administrar Procesadoras



Elaborado por: Carlos Llumiquinga

2.1.2.2.11 Procesar Pedidos De Bouquets

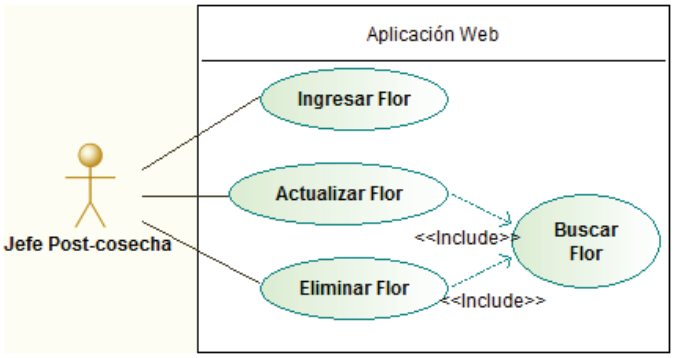
Figura 2.11: Diagrama de CU Procesar Pedidos de Bouquets



Elaborado por: Carlos Llumiquinga

2.1.2.2.12 Administrar Flor

Figura 2.12: Diagrama de CU Administrar Flor



Elaborado por: Carlos Llumiquinga

2.1.2.3 Descripción de los Casos de Uso

En este punto se detallará cada caso de uso de los diagramas anteriores, permitiendo una comprensión más clara de ellos. El detalle se lo realizará por medio de una tabla que contendrá los siguientes campos: Nombre, descripción, actores (principales, secundarios), eventos que lo inician, eventos (primarios, alternativos), pre y postcondiciones.

2.1.2.3.1 Registrar Florícola

La descripción se muestra en la Tabla 2.2.

Tabla 2.2: Descripción CU Registrar Florícola

CASO DE USO	REGISTRAR FLORICOLA
DESCRIPCIÓN: El administrador de la aplicación ingresa los datos de la florícola en la aplicación.	
Actor Principal:	Administrador
Actor secundario:	
Eventos que lo inician:	
1. Seleccionar la opción Administrar. 2. Seleccionar la opción Florícola. 3. La aplicación despliega la información de la florícola si existe.	
Flujo de Eventos Primarios **a. Ingresar Florícola** 1. El Administrador ingresa en la aplicación los campos: 1) Razón Social 2) RUC 3) Dirección 4) Teléfono 5) Página Web 6) Correo Electrónico 2. El Administrador guarda en la aplicación los datos ingresados. 3. La aplicación informa al usuario que los datos fueron guardados con éxito. **Flujo de Eventos Alternativos** 2.1. Datos ingresados incorrectos o incompletos. 2.1.1. La Aplicación solicita al Administrador complete o corrija los datos ingresados. 2.1.2. Administrador corrige los datos solicitados. 2.1.3. Pasa al punto 2 del Evento Primario. 2.2. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal. **b. Modificar Florícola** 1. El Administrador selecciona Modificar Datos. 2. El Administrador modifica los datos necesarios. 3. El Administrador guarda en la aplicación los datos modificados. 4. La Aplicación informa al administrador, que los datos han sido modificados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Administrador complete o corrija los datos ingresados. 3.1.2. Administrador corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal.	

Precondiciones
1. El usuario debe haber ingresado con las credenciales de Administrador.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.2 Administrar Usuarios

La descripción se muestra en la Tabla 2.3.

Tabla 2.3: Descripción CU Administrar Usuarios

CASO DE USO	ADMINISTRAR USUARIOS
DESCRIPCIÓN: El Administrador podrá gestionar los usuarios (ingresar, eliminar o actualizar), que tienen acceso a la aplicación.	
Actor Principal:	Administrador
Actor secundario:	
Eventos que lo inician:	
1. Selecciona la opción Administrar. 2. Selecciona la opción Administrar Usuarios. 3. El sistema despliega una lista con los usuarios existentes.	
Flujo de Eventos Primarios **a. Actualizar Usuario** 1. El Administrador selecciona el Usuario que desea actualizar. 2. El Administrador modifica los datos del Usuario Seleccionado. 3. El Administrador guarda en la Aplicación los datos modificados. 4. La Aplicación informa al Administrador, que los datos han sido actualizados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Administrador complete o corrija los datos ingresados. 3.1.2. Administrador corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal. **b. Ingresar Usuario** 1. El Administrador registra los campos necesarios para un nuevo Usuario: 1) Nombre completo del usuario 2) nombre de usuario 3) contraseña 4) área 5) correo electrónico 6) teléfono. 2. El Administrador guarda los datos ingresados. 3. La Aplicación informa al Administrador, que los datos han sido guardados con éxito.	

Flujo de Eventos Alternativos 2.1. Datos ingresados son incorrectos o incompletos. 2.1.1. La Aplicación solicita al Administrador complete o corrija los datos ingresados. 2.1.2. Administrador corrige los datos solicitados. 2.1.3. Pasa al punto 2 del Evento Primario. 2.2. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal. **c. Eliminar Usuario** 1. El Administrador selecciona el Usuario a eliminar. 2. El Administrador elimina el Usuario seleccionado en la Aplicación. 3. La Aplicación solicita una confirmación para eliminar el Usuario. 4. El Administrador confirma la eliminación del Usuario. 5. La Aplicación informa al Administrador, que el Usuario se eliminó con éxito. **Flujo de Eventos Alternativos** 4.1. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal.
Precondiciones
1. El Usuario debe haber ingresado con las credenciales de Administrador.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.3 Administrar Áreas

La descripción se muestra en la Tabla 2.4.

Tabla 2.4: Descripción CU Administrar Áreas

CASO DE USO	ADMINISTRAR AREAS
DESCRIPCIÓN: El Administrador podrá gestionar las Áreas de la florícola (ingresar, eliminar o actualizar).	
Actor Principal:	Administrador
Actor secundario:	
Eventos que lo inician:	
1. Seleccionar la opción Administrar. 2. Seleccionar la opción Áreas. 3. El sistema despliega una lista con las Áreas existentes.	

Flujo de Eventos Primarios

a. Actualizar Área

1. El Administrador selecciona el Área que desea actualizar.
2. El Administrador modifica los datos del Área Seleccionada.
3. El Administrador guarda en la Aplicación los datos modificados.
4. La Aplicación informa al Administrador, que los datos han sido actualizados con éxito.

Flujo de Eventos Alternativos

3.1. Datos ingresados son incorrectos o incompletos

3.1.1. La Aplicación solicita al Administrador complete o corrija los datos ingresados.

3.1.2. Administrador corrige los datos solicitados.

3.1.3. Pasa al punto 3 del Evento Primario.

3.2. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal.

b. Ingresar Área

1. El Administrador ingresa los campos necesarios para una nueva Área: 1) Nombre 2) Detalle.
2. El Administrador guarda los datos ingresados.
3. La Aplicación informa al Administrador, que los datos han sido guardados con éxito.

Flujo de Eventos Alternativos

2.1. Datos ingresados son incorrectos o incompletos.

2.1.1. La Aplicación solicita al Administrador complete o corrija los datos ingresados.

2.1.2. Administrador corrige los datos solicitados.

2.1.3. Pasa al punto 2 del Evento Primario.

2.2. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal.

c. Eliminar Área

1. El Administrador selecciona el Área a eliminar.
2. El Administrador elimina el Área seleccionado en la Aplicación.
3. La Aplicación solicita una confirmación para eliminar el Área.
4. El Administrador confirma la eliminación del Área.
5. La Aplicación informa al Administrador, que el Área se eliminó con éxito.

Flujo de Eventos Alternativos

4.1. El Administrador selecciona la opción cancelar, la aplicación regresa a la página principal.

4.2 Si el usuario quiere eliminar las Áreas que vienen predefinidas la Aplicación muestra un mensaje diciendo al usuario que no se puede eliminar esa área.

Precondiciones
1. El usuario debe haber ingresado con las credenciales de Administrador.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.4 Administrar Clientes

La descripción se muestra en la Tabla 2.5

Tabla 2.5: Descripción CU Administrar Clientes

CASO DE USO	ADMINISTRAR CLIENTES
DESCRIPCIÓN: El Ejecutivo de Ventas podrá Administrar los Clientes que tenga a cargo (crear, actualizar o eliminar).	
Actor Principal:	Ejecutivo de Ventas (Usuario)
Actor secundario:	Cliente
Eventos que lo inician:	
1. Seleccionar la opción Pedidos. 2. Seleccionar Administrar Clientes. 3. La Aplicación muestra una lista de los Clientes existentes.	
Flujo de Eventos Primarios **a. Actualizar Cliente** 1. El Usuario selecciona el Cliente que desea Actualizar 2. El Usuario modifica los datos del Cliente seleccionado. 3. El Usuario guarda los datos modificados en la Aplicación. 4. La Aplicación informa al Usuario, que los datos han sido actualizados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 3.1.2. El Usuario corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. **b. Ingresar Cliente** 1. El Usuario ingresa los siguientes datos para un nuevo cliente: 1) Razón social, 2) No. Identificación, 3) país, 4) ciudad, 5) dirección, 6) persona de contacto, 7) teléfono convencional, celular, 8) correo electrónico. 2. El Usuario guarda los datos ingresados. 3. La Aplicación informa al Usuario, que los datos han sido guardados con éxito.	

Flujo de Eventos Alternativos 2.1. Datos ingresados son incorrectos o incompletos. 2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 2.1.2. El Usuario corrige los datos solicitados. 2.1.3. Pasa al punto 2 del Evento Primario. 2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. **c. Eliminar Cliente** 1. El Usuario selecciona el Cliente que desea eliminar. 2. El Usuario selecciona eliminar Cliente. 3. La Aplicación solicita una confirmación para eliminar el Cliente. 4. El Usuario confirma la eliminación del Cliente. 5. La Aplicación informa al Usuario, que el Cliente se eliminó con éxito. **Flujo de Eventos Alternativos** 4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.
Precondiciones
1. El Usuario debe haber ingresado con las credenciales de Ejecutivo de Ventas.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.5 Administrar Pedidos De Bouquets

La descripción se muestra en la Tabla 2.6

Tabla 2.6: Descripción CU Administrar Ordenes de Pedidos de Bouquets

CASO DE USO	ADMINISTRAR PEDIDOS DE BOUQUETS
DESCRIPCIÓN: Un cliente realiza un pedido de bouquets a su Ejecutivo de Ventas (vendedor), el cual registra los datos del pedido. También podrá modificar o cancelar un pedido ingresado.	
Actor Principal:	Ejecutivo de Ventas (Vendedor)
Actor secundario:	Cliente
Eventos que lo inician:	
1. El cliente realiza un pedido de bouquets. 2. El Vendedor ingresa a la aplicación con sus credenciales. 3. Selecciona la opción Pedidos. 4. La aplicación muestra una lista de los pedidos realizados, ordenados por la fecha de ingreso.	

Flujo de Eventos Primarios

a. Ingresar Orden

1. El Vendedor ingresa los datos del pedido: 1) El Cliente, 2) fecha salida pedido, 3) lista de bouquets solicitados y la cantidad.
2. El Vendedor guarda los datos del pedido.
3. La Aplicación informa al Vendedor, que los datos han sido guardados con éxito.

Flujo de Eventos Alternativos

2.1. Datos ingresados son incorrectos o incompletos.
2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados.
2.1.2. El Usuario corrige los datos solicitados.
2.1.3. Pasa al punto 2 del Evento Primario.

2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

b. Modificar Orden

1. El Vendedor selecciona el pedido a modificar.
2. El Vendedor modifica los datos del pedido.
3. El Vendedor guarda en la aplicación las modificaciones realizadas.
4. La Aplicación informa al Vendedor, que el pedido fue modificado con éxito.

Flujo de Eventos Alternativos

1.1. Si el estado del pedido es Ingresado se muestra la página para modificar la orden.
1.2 Si el estado del pedido no es Ingresado la aplicación muestra un mensaje indicando que no es posible modificar la orden.

3.1. Datos ingresados son incorrectos o incompletos.
3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados.
3.1.2. El Usuario corrige los datos solicitados.
3.1.3. Pasa al punto 3 del Evento Primario.

3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

c. Cancelar Orden.

1. El vendedor selecciona la orden a cancelar.
2. El Vendedor cancela la orden.
3. La Aplicación solicita confirmar la cancelación de la orden.
4. El Vendedor confirma cancelar la orden.
5. La Aplicación informa al Vendedor la cancelación de la orden.

Flujo de Eventos Alternativos

4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.
4.2. Si el estado del pedido es Ingresado se cancela la orden.

4.2.1 Si el estado del pedido no es Ingresado la aplicación muestra un mensaje indicando que no es posible cancelar la orden.
Precondiciones
1. El Usuario debe ingresar con las credenciales de Vendedor.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.6 Administrar Bouquets

La descripción se muestra en la Tabla 2.7.

Tabla 2.7: Descripción CU Administrar Bouquets

CASO DE USO	ADMINISTRAR BOUQUETS
DESCRIPCIÓN: El Jefe de Bouquetera podrá gestionar los bouquets y sus respectivas recetas (ingresar, eliminar o actualizar).	
Actor Principal:	Jefe de Bouquetera (Usuario)
Actor secundario:	
Eventos que lo inician:	
1. El Usuario Selecciona la opción Bouquets. 2. El Usuario Selecciona Administrar. 3. La Aplicación muestra una lista de los bouquets disponibles.	
Flujo de Eventos Primarios **a. Editar Bouquet** 1. El Usuario selecciona el bouquet que desea modificar. 2. El Usuario modifica los datos del bouquet seleccionado. 3. El Usuario guarda en la Aplicación los datos modificados. 4. La Aplicación informa al Usuario, que los datos han sido modificados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 3.1.2. El Usuario corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. **b. Ingresar Bouquet** 1. El Usuario ingresa los campos necesarios para un Nuevo Bouquet y su receta genérica: 1) Nombre del Bouquet 2) Ocasión 3) Medidas 4) No. Flores focales 5) No. Flores de línea 6) No. Flores de Relleno 7) No. Verdes 8) Accesorios 9) Fotografía 10) Detalles. 2. El Usuario guarda los datos ingresados.	

3. La Aplicación informa al Usuario, que los datos han sido guardados con éxito.
4. Se muestra la página para ingresar los detalles de la receta del bouquet: 1)Flores Focales 2) Flores de Línea 3) Flores de Relleno 4)Flores Verdes 5) Accesorios 6) Indicaciones
5. El usuario selecciona el tipo de flor, busca la flor e ingresa la cantidad a usar.
6. El usuario selecciona editar indicaciones.
7. El usuario puede cerrar la página o imprimir la receta del bouquet.

Flujo de Eventos Alternativos

2.1. Datos ingresados son incorrectos o incompletos
2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados.
2.1.2. El Usuario corrige los datos solicitados.
2.1.3. Pasa al punto 2 del Evento Primario.
2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

5.1. La cantidad de flores ingresada supera al valor ingresado en la receta genérica.
5.1.1. La aplicación informa al usuario que la cantidad ingresada es incorrecta.
5.1.2. El usuario corrige la cantidad de flor ingresada.
5.1.3. Pasa al punto 5 del Evento Primario.

c. Ver Receta

1. Se muestra la página para ingresar los detalles de la receta del bouquet: 1)Flores Focales 2) Flores de Línea 3) Flores de Relleno 4)Flores Verdes 5) Accesorios 6) Indicaciones
2. El usuario selecciona el tipo de flor, busca la flor e ingresa la cantidad a usar.
3. El usuario selecciona el ítem del tipo de flor y puede modificar la cantidad registrada.
4. El usuario selecciona editar indicaciones.
5. El usuario puede cerrar la página o imprimir la receta del bouquet.

Flujo de Eventos Alternativos

2.1. La cantidad de flores ingresada supera al valor ingresado en la receta genérica.
2.1.1. La aplicación informa al usuario que la cantidad ingresada es incorrecta.
2.1.3. Pasa al punto 2 del Evento Primario.

d. Eliminar Bouquet

6. El Usuario busca el bouquet a eliminar.
7. El Usuario elimina el bouquet.
8. La Aplicación solicita una confirmación para eliminar el bouquet.
9. El Usuario confirma la eliminación del bouquet.

10. La Aplicación informa al Usuario, que el bouquet se eliminó con éxito. **Flujo de Eventos Alternativos** 4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.
Precondiciones
1. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.7 Administrar Accesorios

La descripción se muestra en la Tabla 2.8.

Tabla 2.8: Descripción CU Administrar Accesorios

CASO DE USO	ADMINISTRAR ACCESORIOS
DESCRIPCIÓN: El Jefe de Bouquetera podrá gestionar los Accesorios que se utilizarán para elaborar los bouquets (ingresar, eliminar o actualizar).	
Actor Principal:	Jefe de Bouquetera (Usuario)
Actor secundario:	
Eventos que lo inician:	
1. El Usuario Selecciona la opción Bouquets. 2. El Usuario Selecciona Administrar Accesorios. 3. La aplicación despliega una lista con los Accesorios disponibles.	
Flujo de Eventos Primarios **a. Actualizar Accesorio** 1. El Usuario selecciona el Accesorio que desea actualizar. 2. El Usuario modifica los datos del Accesorio Seleccionado. 3. El Usuario guarda en la Aplicación los datos modificados. 4. La Aplicación informa al Usuario, que los datos han sido actualizados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 3.1.2. El Usuario corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.	

b. Ingresar Accesorio 1. El Usuario ingresa los campos necesarios para un Nuevo Accesorio: 1) Nombre del Accesorio 2) Fotografía 3) Descripción. 2. El Usuario guarda los datos ingresados. 3. La Aplicación informa al Usuario, que los datos han sido guardados con éxito. **Flujo de Eventos Alternativos** 2.1. Datos ingresados son incorrectos o incompletos 2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 2.1.2. El Usuario corrige los datos solicitados. 2.1.3. Pasa al punto 2 del Evento Primario. 2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. **c. Eliminar Accesorio** 1. El Usuario selecciona el Accesorio a eliminar. 2. El Usuario elimina el Accesorio seleccionado. 3. La Aplicación solicita una confirmación para eliminar el Accesorio. 4. El Usuario confirma la eliminación del Accesorio. 5. La Aplicación informa al Usuario, que el Accesorio se eliminó con éxito. **Flujo de Eventos Alternativos** 4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. 4.2 Si el Accesorio consta en la receta de un Bouquet la aplicación muestra un mensaje indicando que no se puede eliminar.
Precondiciones
1. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.8 Administrar Ocasión Bouquet

La descripción se muestra en la Tabla 2.9.

Tabla 2.9: Descripción CU Administrar Ocasión Bouquet

CASO DE USO	ADMINISTRAR OCASIÓN BOUQUET
DESCRIPCIÓN: El Jefe de Bouquetera podrá administrar las Ocasiones para los que se elaboraran los bouquets (ingresar, eliminar o actualizar).	
Actor Principal:	Jefe de Bouquetera (Usuario)
Actor secundario:	
Eventos que lo inician:	
1. El Usuario Selecciona la opción Bouquets. 2. El Usuario Selecciona Administrar Ocasión Bouquet. 3. La aplicación despliega una lista con las Ocasiones disponibles.	
Flujo de Eventos Primarios **d. Actualizar Accesorio** 1. El Usuario selecciona la Ocasión que desea actualizar. 2. El Usuario modifica los datos de la Ocasión Seleccionada. 3. El Usuario guarda en la Aplicación los datos modificados. 4. La Aplicación informa al Usuario, que los datos han sido actualizados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 3.1.2. El Usuario corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. **e. Ingresar Ocasión** 1. El Usuario ingresa los campos necesarios para una Nueva Ocasión: 1) Nombre de la Ocasión 2) Descripción. 2. El Usuario guarda los datos ingresados. 3. La Aplicación informa al Usuario, que los datos han sido guardados con éxito. **Flujo de Eventos Alternativos** 2.1. Datos ingresados son incorrectos o incompletos 2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 2.1.2. El Usuario corrige los datos solicitados. 2.1.3. Pasa al punto 2 del Evento Primario. 2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.	

f. Eliminar Accesorio 1. El Usuario selecciona la Ocasión a eliminar. 2. El Usuario elimina la Ocasión seleccionada. 3. La Aplicación solicita una confirmación para eliminar la Ocasión. 4. El Usuario confirma la eliminación de la Ocasión. 5. La Aplicación informa al Usuario, que la Ocasión se eliminó con éxito. **Flujo de Eventos Alternativos** 4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.
Precondiciones
1. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.9 Administrar Procesadoras

La descripción se muestra en la Tabla 2.10.

Tabla 2.10: Descripción CU Administrar Procesadoras

CASO DE USO	ADMINISTRAR PROCESADORAS
DESCRIPCIÓN: El Jefe de Bouquetera podrá Administrar las procesadoras (ingresar, eliminar o actualizar), que se encargan de elaborar los bouquets.	
Actor Principal:	Jefe de Bouquetera (Usuario)
Actor secundario:	
Eventos que lo inician:	
1. Selecciona la opción Bouquets. 2. Selecciona la opción Administrar Procesadoras. 3. La aplicación despliega una lista de las procesadoras ingresadas.	
Flujo de Eventos Primarios **a. Actualizar Procesadora** 1. El Usuario selecciona la Procesadora que desea actualizar. 2. El Usuario modifica los datos de la Procesadora seleccionada. 3. El Usuario guarda en la Aplicación los datos modificados. 4. La Aplicación informa al Usuario, que los datos han sido actualizados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 3.1.2. El Usuario corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario.	

3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

Flujo de Eventos Alternativos

a. Ingresar Procesadora

1. El Usuario registra los campos necesarios para la nueva Procesadora: 1) Nombre completo 2) cedula de identidad 3) fecha de nacimiento 4) dirección domicilio 5) teléfono 6) observaciones.
2. El Usuario guarda los datos ingresados.
3. La Aplicación informa al Usuario, que los datos han sido guardados con éxito.

Flujo de Eventos Alternativos

2.1. Datos ingresados son incorrectos o incompletos

2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados.

2.1.2. El Usuario corrige los datos solicitados.

2.1.3. Pasa al punto 3 del Evento Primario.

2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

b. Eliminar Procesadora

1. El Usuario selecciona la Procesadora a eliminar.
2. El Usuario elimina la Procesadora seleccionada.
3. La Aplicación solicita una confirmación para eliminar la Procesadora.
4. El Usuario confirma la eliminación de la Procesadora.
5. La Aplicación informa al Usuario, que la Procesadora se eliminó con éxito.

Flujo de Eventos Alternativos

4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

Precondiciones

1. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.

Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.9.1 Solicitar Flor y Accesorios

La descripción se muestra en la Tabla 2.11.

Tabla 2.11: Descripción CU Solicitar Flor y Accesorios

CASO DE USO	SOLICITAR FLOR Y ACCESORIOS
DESCRIPCIÓN: El jefe de Bouquetera genera un documento con la cantidad de tipo de flor necesaria para elaborar los bouquets y sus respectivos accesorios para una fecha de salida de los pedidos.	
Actor Principal:	Jefe de Bouquetera (Usuario)
Actor secundario:	Procesadora
Eventos que lo inician: 1. Selecciona la opción Bouquets. 2. Selecciona la opción Procesar Pedidos. 3. Selecciona la opción Solicitar Flor y Accesorios.	
Flujo de Eventos Primarios 1. Se muestra una lista de órdenes de pedido con fecha al siguiente día del actual, donde se obtendrán las cantidades de flores y accesorios. 2. El Usuario genera el documento de solicitud de flor y accesorios. 3. El Usuario imprime o guarda el documento con la cantidad y tipos de flores necesarias para elaborar las órdenes que serán enviadas a la postcosecha además de los accesorios necesarios.	
Precondiciones	
1. La fecha de salida de las órdenes es el día siguiente a la fecha actual. 2. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.	
Postcondiciones	

Elaborado por: Carlos Llumiquinga

2.1.2.3.9.2 Asignar Elaboración de Bouquets.

La descripción se muestra en la Tabla 2.12.

Tabla 2.12: Descripción CU Asignar Elaboración de Bouquets

CASO DE USO	ASIGNAR ELABORACION DE BOUQUETS
DESCRIPCIÓN: El jefe de Bouquetera de acuerdo a la fecha de salida de las órdenes, procede a realizar la asignación de los bouquets a las procesadoras para su elaboración.	
Actor Principal:	Jefe de Bouquetera (Usuario)
Actor secundario:	Procesadora

Eventos que lo inician:

1. Seleccionar la opción Bouquets.
2. Selecciona la opción Procesar Pedidos.
3. Selecciona la opción Asignar Elaboración.
4. Se muestra una lista de los bouquets y la cantidad a elaborarse.

Flujo de Eventos Primarios

a. Seleccionar Bouquet

1. El Usuario selecciona el bouquet a asignar.
2. Se muestra el detalle del bouquet y una lista de los clientes y cantidad del bouquet que solicitaron.

b. Ver Asignaciones

1. El Usuario selecciona ver asignaciones.
2. Se muestra una ventana con todos los bouquets asignados a las procesadoras.
3. El usuario selecciona la procesadora a quien se le quitará la asignación.
4. Se muestra una lista con los bouquets asignados a la procesadora
5. El usuario elimina una asignación de elaboración.

Flujo de Eventos Alternativos

5.1 El usuario puede seleccionar el bouquet asignado para imprimir la etiqueta en caso de ser necesario.
5.1.2 Seleccionar el formato de impresión.
5.1.3 Imprimir las etiquetas seleccionadas.

Flujo de Eventos Alternativos

a. Asignar Bouquet para elaboración

1. El Usuario ingresa la cantidad, selecciona la procesadora y asigna la elaboración, hasta asignar por completo la cantidad total del bouquet.
2. Una vez completada la asignación de bouquets para su elaboración, se muestra la opción de imprimir las asignaciones realizadas.
3. El usuario selecciona Ver Asignados, donde podrá editar las asignaciones para elaboración.

Flujo de Eventos Alternativos

1.1. La cantidad asignada supera el total de bouquets a elaborar.
1.1.1. La Aplicación muestra un mensaje indicando al usuario que la cantidad ingresada excede al total de bouquets a elaborar.
1.1.2. El Usuario corrige la cantidad asignada.
1.1.3. Pasa al punto 1 del Evento Primario.

1.2. El usuario selecciona Regresar, para volver a la lista de bouquets para asignar su elaboración.

Precondiciones
1. Haber realizado la Solicitud de flor y accesorios para la fecha de elaboración. 2. Información y detalle de las órdenes. 3. Disponer de la flor necesaria para la elaboración de los bouquets. 4. Disponer del personal necesario de procesadoras. 5. La fecha de salida de las órdenes no puede ser menor o igual a la fecha actual. 6. Una asignación puede ser eliminada si aún no se ha registrado su elaboración. 7. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.
Postcondiciones
Si no se cuenta con el personal necesario se debe extender el horario de trabajo, o solicitar el personal de otras áreas.

Elaborado por: Carlos Llumiquinga

2.1.2.3.9.3 Imprimir etiquetas.

La descripción se muestra en la Tabla 2.13.

Tabla 2.13: Descripción CU Imprimir Etiquetas

CASO DE USO	IMPRIMIR ETIQUETAS
DESCRIPCIÓN: El jefe de Bouquetera imprimirá las etiquetas de los bouquets asignados a cada procesadora.	
Actor Principal:	Jefe de Bouquetera
Actor secundario:	Procesadora
Eventos que lo inician: 1. Seleccionar la opción Bouquets. 2. Seleccionar la opción Procesar Pedidos. 3. Seleccionar el botón Imprimir Etiquetas.	
Flujo de Eventos Primarios 1. La aplicación muestra una página donde el usuario debe seleccionar el formato de impresión de las etiquetas y la procesadora. 2. Selecciona el formato. 3. Selecciona una procesadora 4. Selecciona la opción imprimir. 5. Se imprimen las etiquetas que contienen: Código de Elaboración, nombre de la Florícola, nombre del bouquet, nombre del cliente, dirección del cliente.	

Precondiciones
1. Todos los bouquets a elaborarse en la fecha actual debieron ser asignados a las procesadoras. 2. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera. 3. Haber asignado a las procesadoras los bouquets de las órdenes que van a elaborar.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.9.4 Asignar Flores y Accesorios.

La descripción se muestra en la Tabla 2.14.

Tabla 2.14: Descripción CU Asignar Flores y Accesorios

CASO DE USO	ASIGNAR FLORES Y ACCESORIOS
DESCRIPCIÓN: El jefe de Bouquetera imprimirá un documento con la respectiva cantidad de flores y accesorios que serán asignados a las procesadoras para elaborar los bouquets.	
Actor Principal:	Jefe de Bouquetera
Actor secundario:	Procesadora
Eventos que lo inician: 1. Seleccionar la opción Bouquets. 2. Seleccionar la opción Procesar Pedidos. 3. Seleccionar el botón Asignar Flores y Accesorios. 4.	
Flujo de Eventos Primarios 1. La Aplicación despliega una ventana con el detalle por procesadoras y la cantidad de flores y accesorios que les serán entregados. 2. Selecciona la opción imprimir. 3. Se imprime, la cantidad, tipo de flor y los accesorios necesarios para la elaboración de los bouquets que fueron asignados a las procesadoras.	
Precondiciones	
1. Todos los bouquets a elaborarse en la fecha actual debieron ser asignados a las procesadoras. 2. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.	
Postcondiciones	

Elaborado por: Carlos Llumiquinga

2.1.2.3.9.5 Registrar Bouquet elaborado

La descripción se muestra en la Tabla 2.15.

Tabla 2.15: Descripción CU Registrar Bouquet Elaborado

CASO DE USO	REGISTRAR BOUQUET ELABORADO
DESCRIPCIÓN: El bouquet al terminar de ser elaborado pasa por un control de calidad, donde se ingresan observaciones si las hay, acerca de su elaboración por parte de la procesadora y se da por finalizado su proceso.	
Actor Principal:	Jefe de Bouquetera
Actor secundario:	Digitador
Eventos que lo inician: 1. Seleccionar la opción Bouquets. 2. Seleccionar la opción Procesar Pedidos. 3. Seleccionar la opción Registrar Elaboración.	
Flujo de Eventos Primarios 1. El Usuario selecciona la procesadora. 2. El Usuario selecciona el bouquet elaborado. 3. Ingresa las observaciones correspondientes. 4. El Usuario Selecciona la opción guardar. 5. Se imprime una etiqueta con los siguientes detalles: Orden, Cliente, Bouquet	
Precondiciones	
1. Haber asignado los bouquets. 2. El Usuario debe haber ingresado con las credenciales de Jefe de Bouquetera.	
Postcondiciones	

Elaborado por: Carlos Llumiquinga

2.1.2.3.10 Administrar Flor

La descripción se muestra en la Tabla 2.16.

Tabla 2.16: Descripción CU Administrar Flor

CASO DE USO	ADMINISTRAR FLOR
DESCRIPCIÓN: El Jefe de Postcosecha podrá gestionar las flores (ingresar, eliminar o actualizar) disponibles en la florícola.	
Actor Principal:	Jefe de Bouquetera (Usuario)
Actor secundario:	
Eventos que lo inician:	
1. El Usuario Selecciona la opción Flor. 2. El Usuario Selecciona Administrar.	
Flujo de Eventos Primarios **a. Actualizar Flor** 1. El Usuario selecciona la Flor que desea actualizar. 2. El Usuario modifica los datos de la Flor Seleccionada. 3. El Usuario guarda en la Aplicación los datos modificados. 4. La Aplicación informa al Usuario, que los datos han sido actualizados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 3.1.2. Usuario corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. **b. Ingresar Flor** 1. El Usuario ingresa los campos necesarios para una Nueva Flor: 1) Nombre de la flor 2) Tipo de Flor 3) Fotografía 4) Descripción. 2. El Usuario guarda los datos ingresados. 3. La Aplicación informa al Usuario, que los datos han sido guardados con éxito. **Flujo de Eventos Alternativos** 2.1. Datos ingresados son incorrectos o incompletos 2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 2.1.2. Usuario corrige los datos solicitados. 2.1.3. Pasa al punto 2 del Evento Primario. 2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. **c. Eliminar Flor** 1. El Usuario selecciona la Flor a eliminar. 2. El Usuario elimina la Flor seleccionada.	

3. La Aplicación solicita una confirmación para eliminar la Flor. 4. El Usuario confirma la eliminación de la Flor. 5. La Aplicación informa al Usuario, que la Flor se eliminó con éxito. **Flujo de Eventos Alternativos** 4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal. 4.2 Si la Flor consta en la receta de un Bouquet la aplicación muestra un mensaje indicando que no se puede eliminar.
Precondiciones
1. El Usuario debe haber ingresado con las credenciales de Jefe de PostCosecha.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.11 Administrar Tipo Flor

La descripción se muestra en la Tabla 2.17.

Tabla 2.17: Descripción CU Administrar Tipo Flor

CASO DE USO	ADMINISTRAR TIPO FLOR
DESCRIPCIÓN: El Jefe de Postcosecha podrá gestionar los Tipos de Flores (ingresar, eliminar o actualizar) disponibles en la florícola.	
Actor Principal:	Jefe de PostCosecha (Usuario)
Actor secundario:	
Eventos que lo inician: 1. Seleccionar la opción PostCosecha. 2. Seleccionar la opción Tipo de Flor.	
Flujo de Eventos Primarios 1. El Usuario selecciona el Tipo de Flor que desea actualizar. 2. El Usuario modifica los datos del Tipo de Flor Seleccionada. 3. El Usuario guarda en la Aplicación los datos modificados. 4. La Aplicación informa al Usuario, que los datos han sido actualizados con éxito. **Flujo de Eventos Alternativos** 3.1. Datos ingresados son incorrectos o incompletos 3.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados. 3.1.2. Usuario corrige los datos solicitados. 3.1.3. Pasa al punto 3 del Evento Primario. 3.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.	

d. Ingresar Tipo de Flor

1. El Usuario ingresa los campos necesarios para un Nuevo Tipo de Flor: 1) Nombre de la flor 2) Descripción.
2. El Usuario guarda los datos ingresados.
3. La Aplicación informa al Usuario, que los datos han sido guardados con éxito.

Flujo de Eventos Alternativos

2.1. Datos ingresados son incorrectos o incompletos

2.1.1. La Aplicación solicita al Usuario complete o corrija los datos ingresados.

2.1.2. Usuario corrige los datos solicitados.

2.1.3. Pasa al punto 2 del Evento Primario.

2.2. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

e. Eliminar Tipo de Flor

1. El Usuario selecciona el Tipo de Flor a eliminar.
2. El Usuario elimina el Tipo de Flor seleccionada.
3. La Aplicación solicita una confirmación para eliminar el Tipo de Flor.
4. El Usuario confirma la eliminación.
5. La Aplicación confirma al Usuario, que el Tipo de Flor se eliminó con éxito.

Flujo de Eventos Alternativos

3.1. Si el Tipo de Flor es un registro inicial de la aplicación se informa al usuario que no se puede eliminar.

4.1. El Usuario selecciona la opción cancelar, la aplicación regresa a la página principal.

Precondiciones
1. El Usuario debe haber ingresado con las credenciales de Jefe de PostCosecha.
Postcondiciones

Elaborado por: Carlos Llumiquinga

2.1.2.3.12 Generar Reportes

La descripción se muestra en la Tabla 2.18.

Tabla 2.18: Descripción CU Generar Reportes

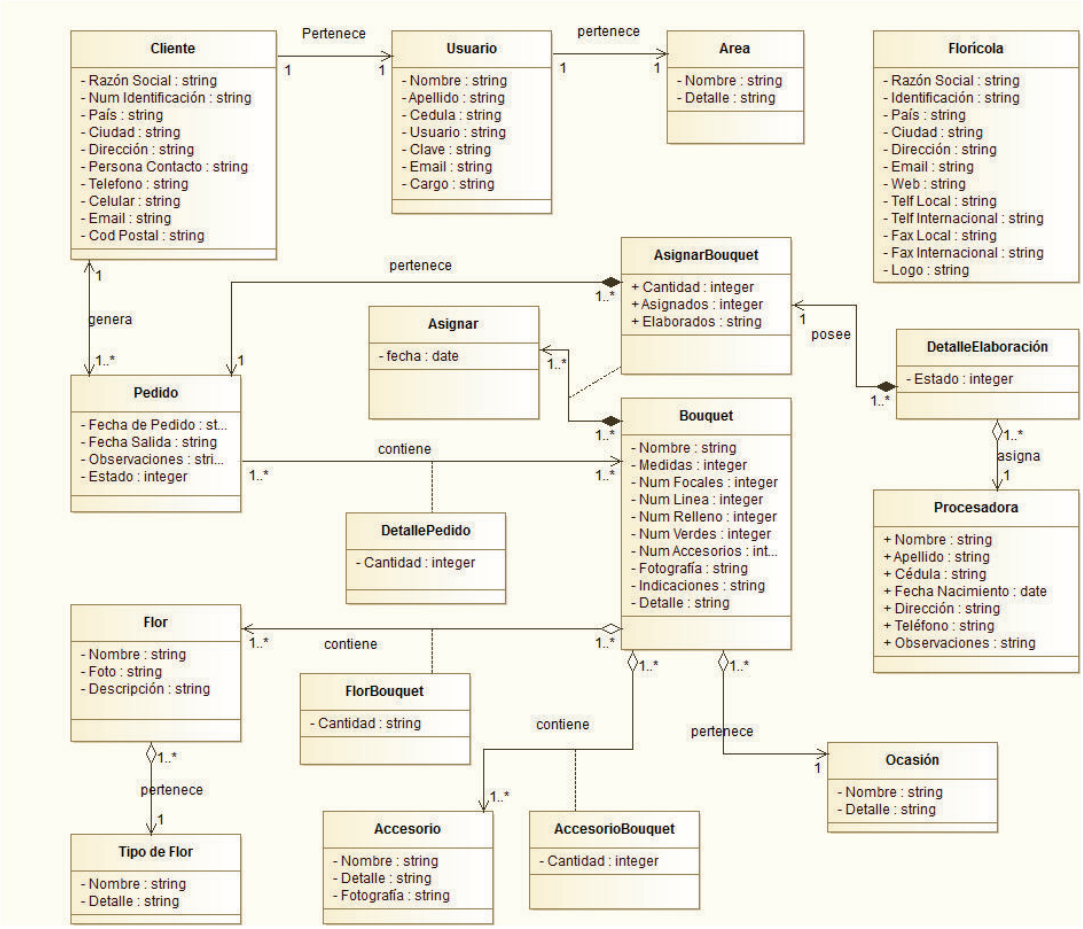
CASO DE USO	GENERAR REPORTES
DESCRIPCIÓN: La Aplicación permite generar reportes de acuerdo a la información generada en el proceso de elaboración de bouquets.	
Actor Principal:	Jefe de Bouquetera, Gerente, Ejecutivo de Ventas
Actor secundario:	
Eventos que lo inician: 1. Seleccionar la opción Reportes.	
Flujo de Eventos Primarios	
1. El Usuario selecciona el tipo de reporte deseado. 2. La Aplicación muestra el reporte seleccionado. 3. El Usuario selecciona el formato del reporte 4. El Usuario guarda o imprime el reporte.	
Flujo de Eventos Alternativos	
Precondiciones	
Postcondiciones	

Elaborado por: Carlos Llumiquinga

2.1.3 MODELO DE CONTENIDO

Representado por un diagrama UML de clases se muestra en la Figura 2.13. Donde constan las principales clases de la aplicación y las relaciones existentes entre ellas. El diagrama no contiene las operaciones o métodos dentro de las clases, ya que la base de datos que se utilizará no es orientada a objetos.

Figura 2.13: Diagrama de Clases



Elaborado por: Carlos Llumiquinga

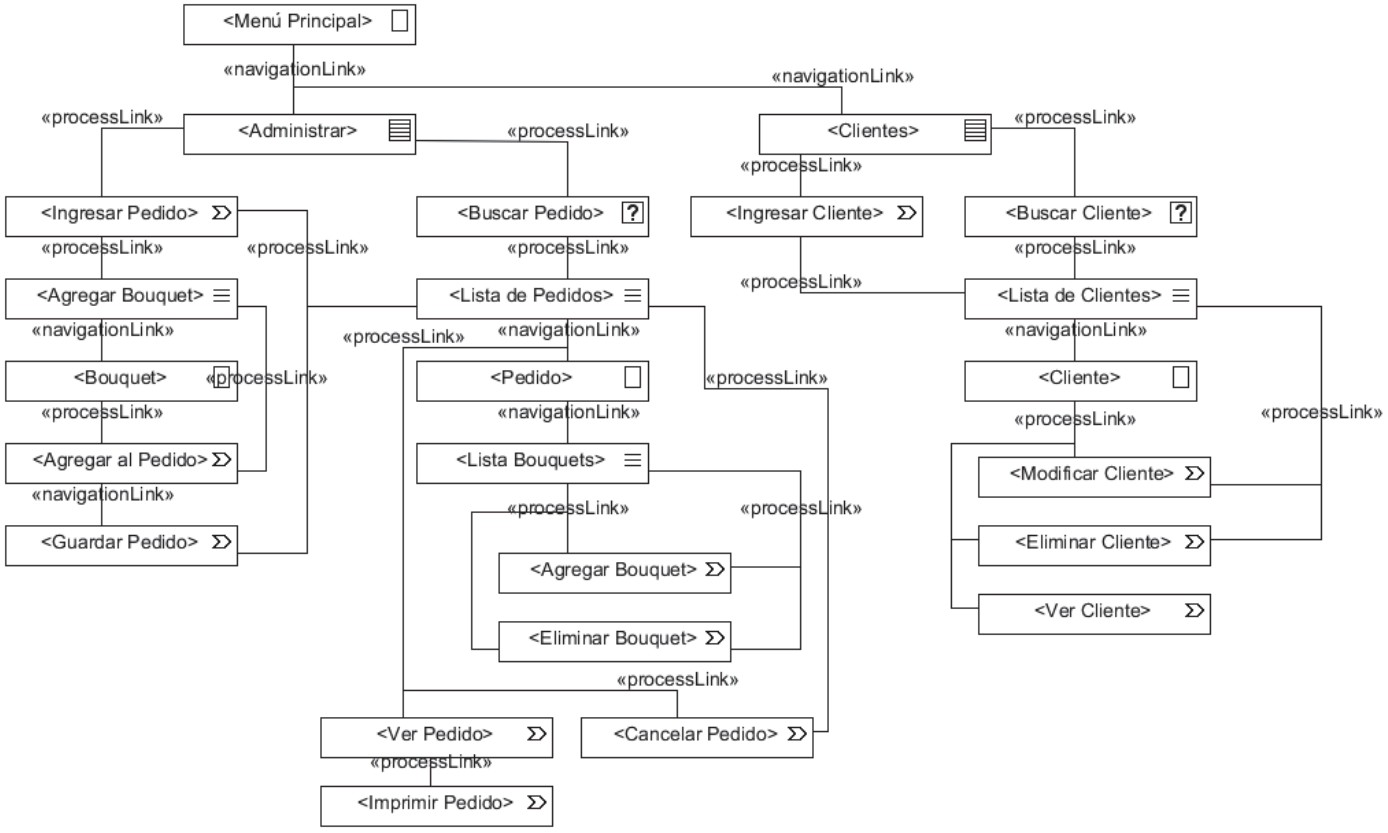
2.1.4 MODELO DE NAVEGACIÓN

En este modelo se presentan los diagramas de navegación para los módulos que contendrán las clases relevantes de la aplicación: 1) Pedidos, 2) Bouquets, 3) Procesar Pedidos ,4) Flor, 5) Administrar, 7) Reportes.

2.1.4.1 Navegación Pedidos

El diagrama se muestra en la Figura 2.14.

Figura 2.14: Diagrama de Navegación Pedidos

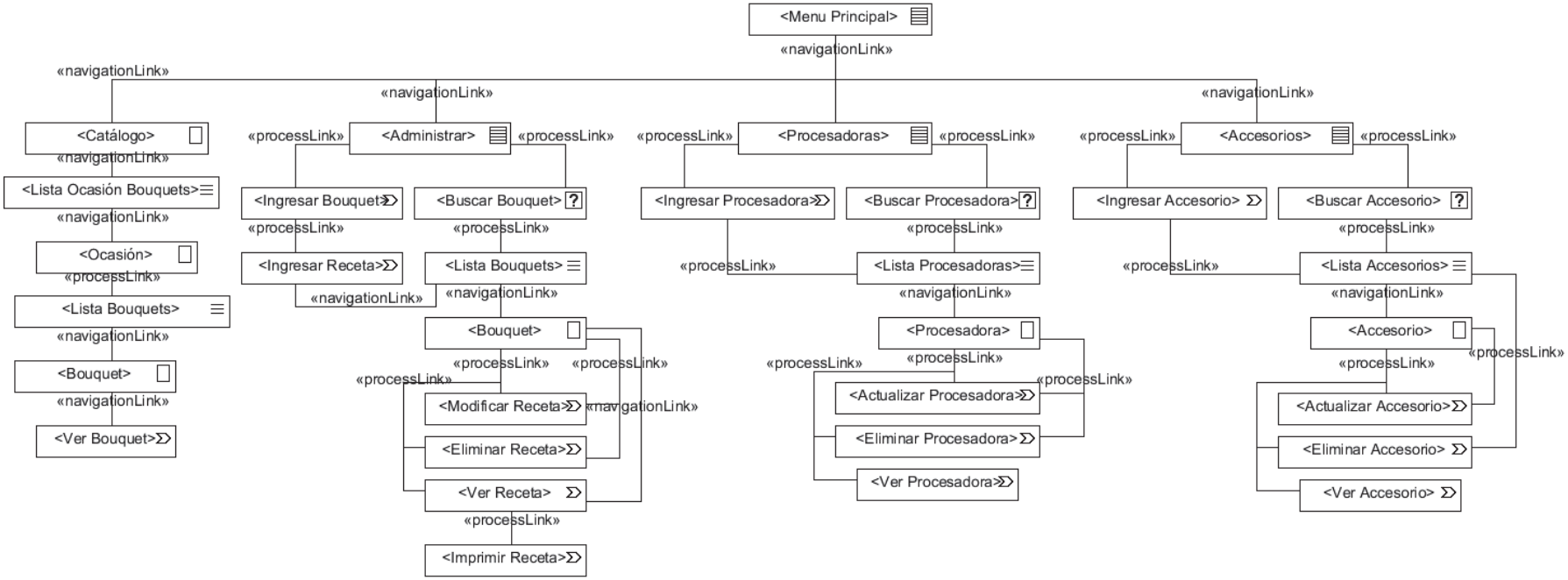


Elaborado por: Carlos Llumiquinga

2.1.4.2 Navegación Bouquets

El diagrama se muestra en la Figura 2.15.

Figura 2.15: Diagrama de Navegación Bouquets

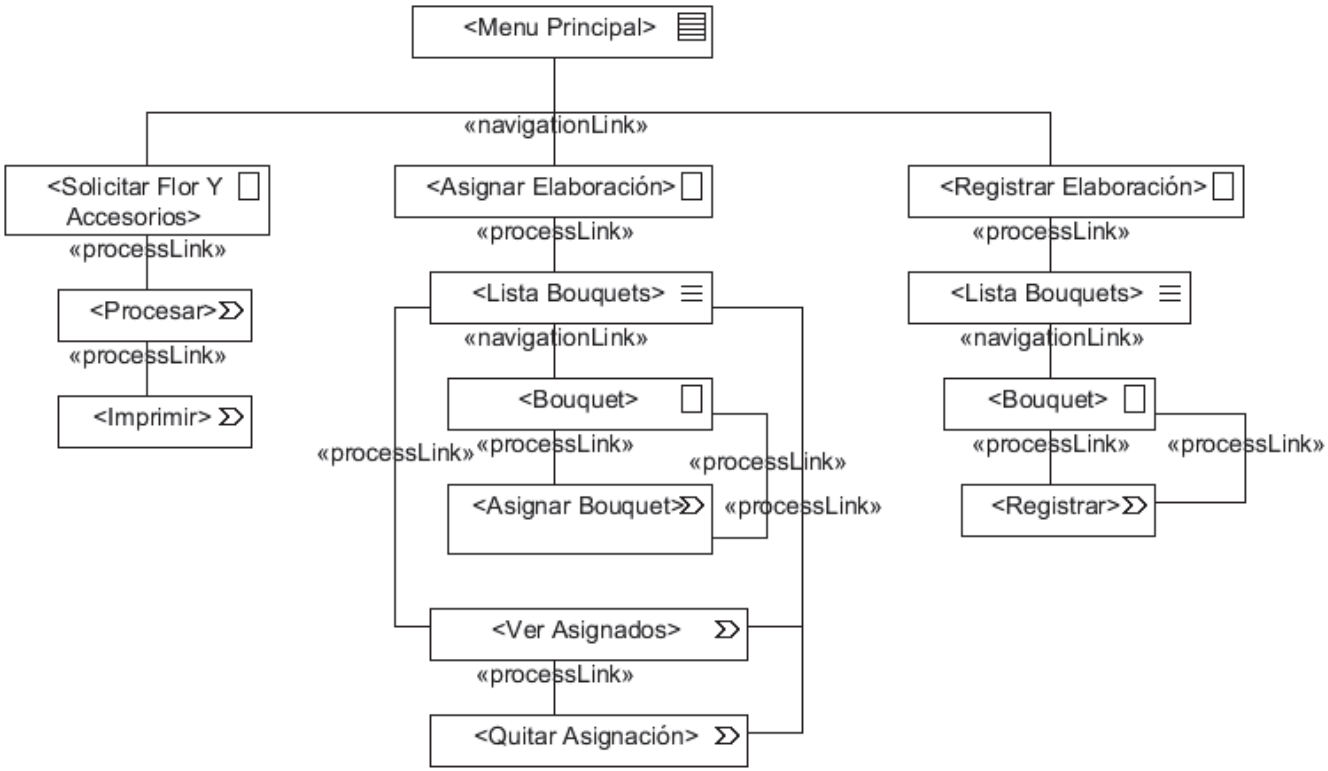


Elaborado por: Carlos Llumiquinga

2.1.4.3 Navegación Procesar Pedidos

El diagrama se muestra en la Figura 2.16.

Figura 2.16: Diagrama de Navegación Procesar Pedidos

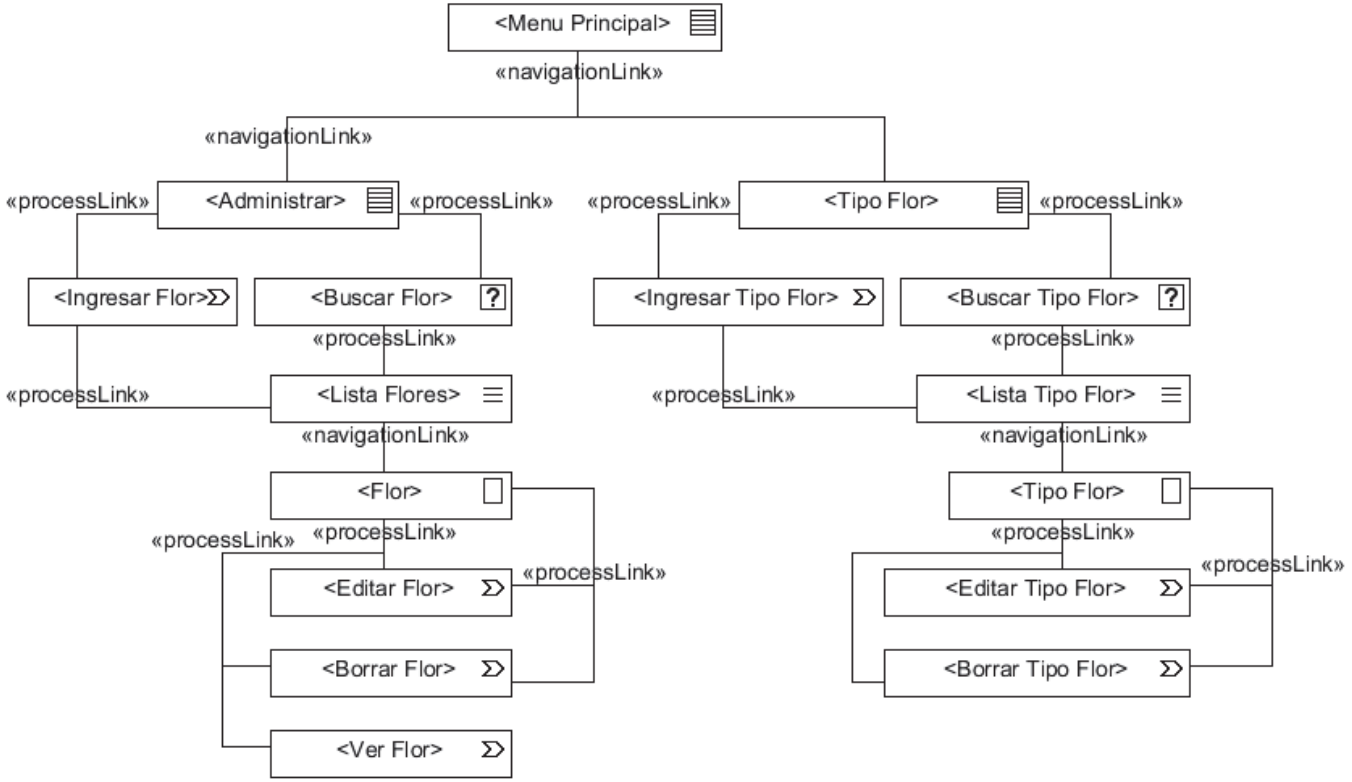


Elaborado por: Carlos Llumiquinga

2.1.4.4 Navegación Flor

El diagrama se muestra en la Figura 2.17.

Figura 2.17: Diagrama de Navegación Flor

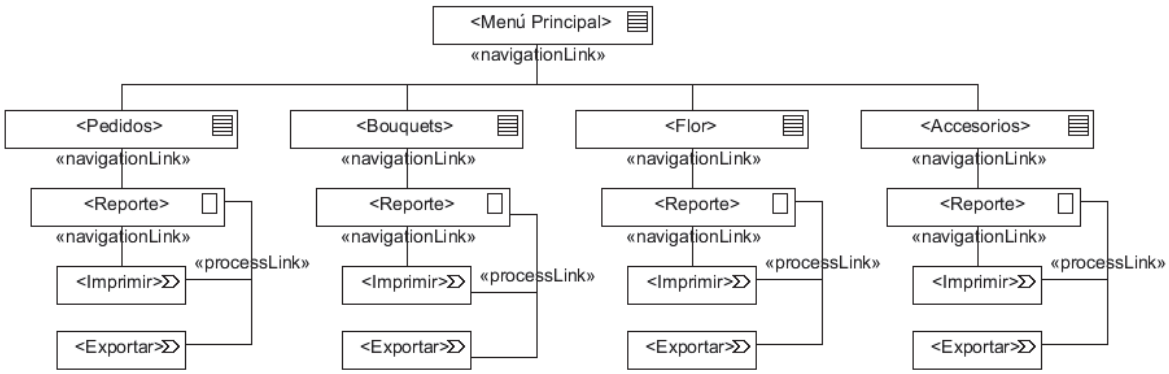


Elaborado por: Carlos Llumiquinga

2.1.4.5 Navegación Reportes

El diagrama se muestra en la Figura 2.18.

Figura 2.18: Diagrama de Navegación Reportes

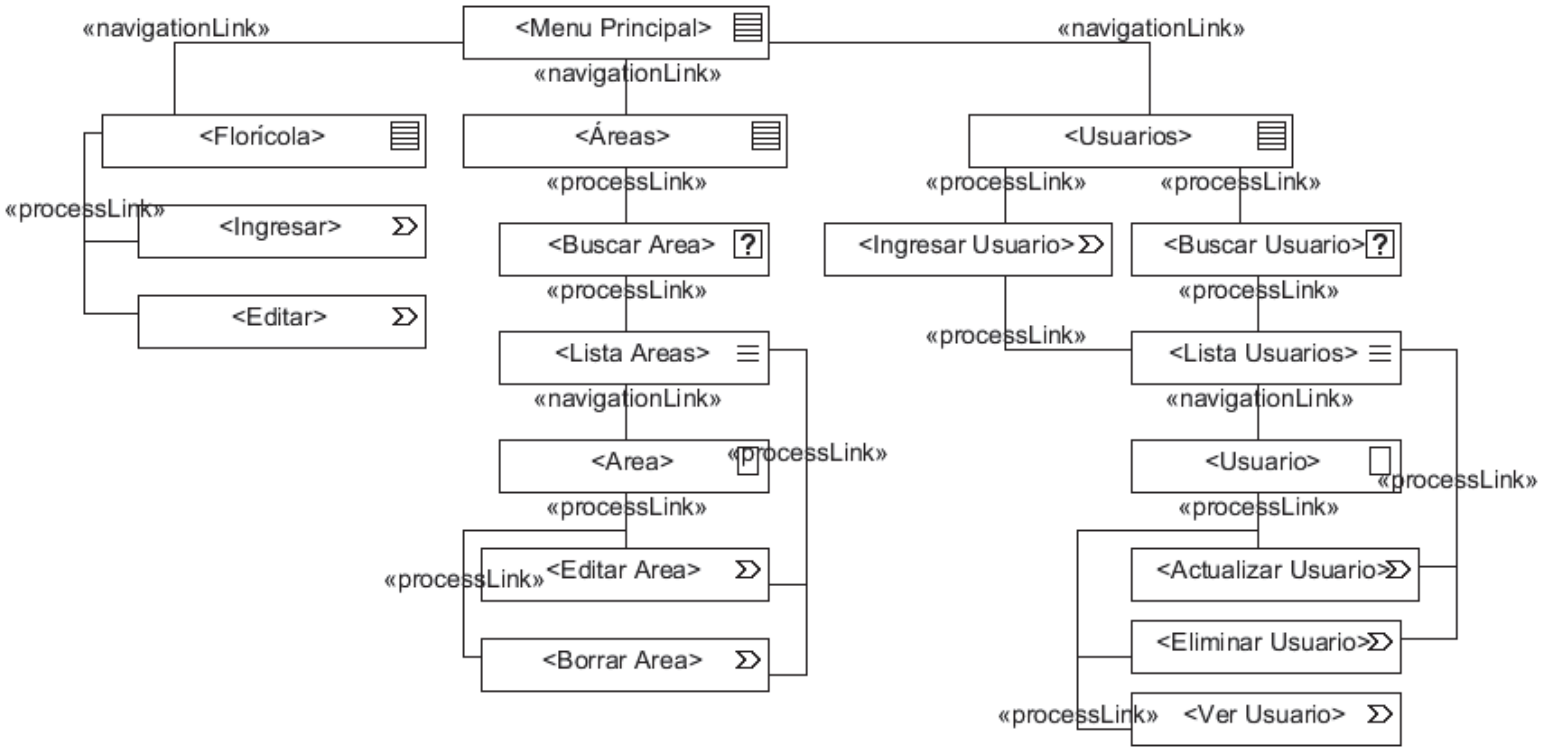


Elaborado por: Carlos Llumiquinga

2.1.4.6 Navegación Administrar

El diagrama se muestra en la Figura 2.19.

Figura 2.19: Diagrama de Navegación Administrar



Elaborado por: Carlos Llumiquinga

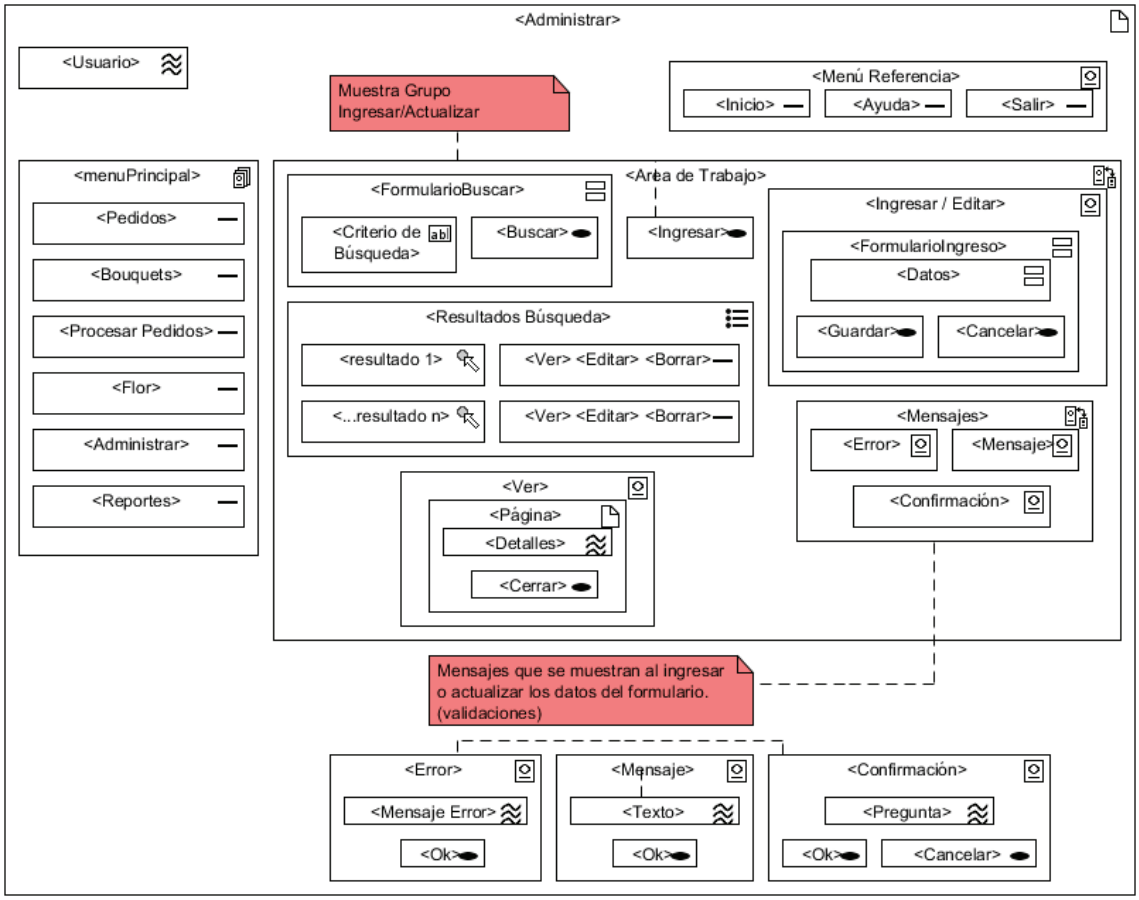
2.1.5 MODELO DE PRESENTACION

Los modelos de presentación nos servirán para poder identificar que clases de navegación y proceso revisadas en el punto anterior pertenecen a la página web. UWE utiliza los diagramas de presentación.

2.1.5.1 Modelo de presentación Administrar

El modelo de presentación administrar será una plantilla para la gestión de las clases básicas dentro de la aplicación, como lo son: Usuario, Área, Clientes, Accesorios, Procesadoras, Ocasión, Flor y Tipo Flor, que presentan una disposición similar en sus elementos dentro de la página web y varían en los elementos correspondientes, que estarán contenidos dentro del estereotipo formulario.

Figura 2.20: Diagrama de presentación para Administrar



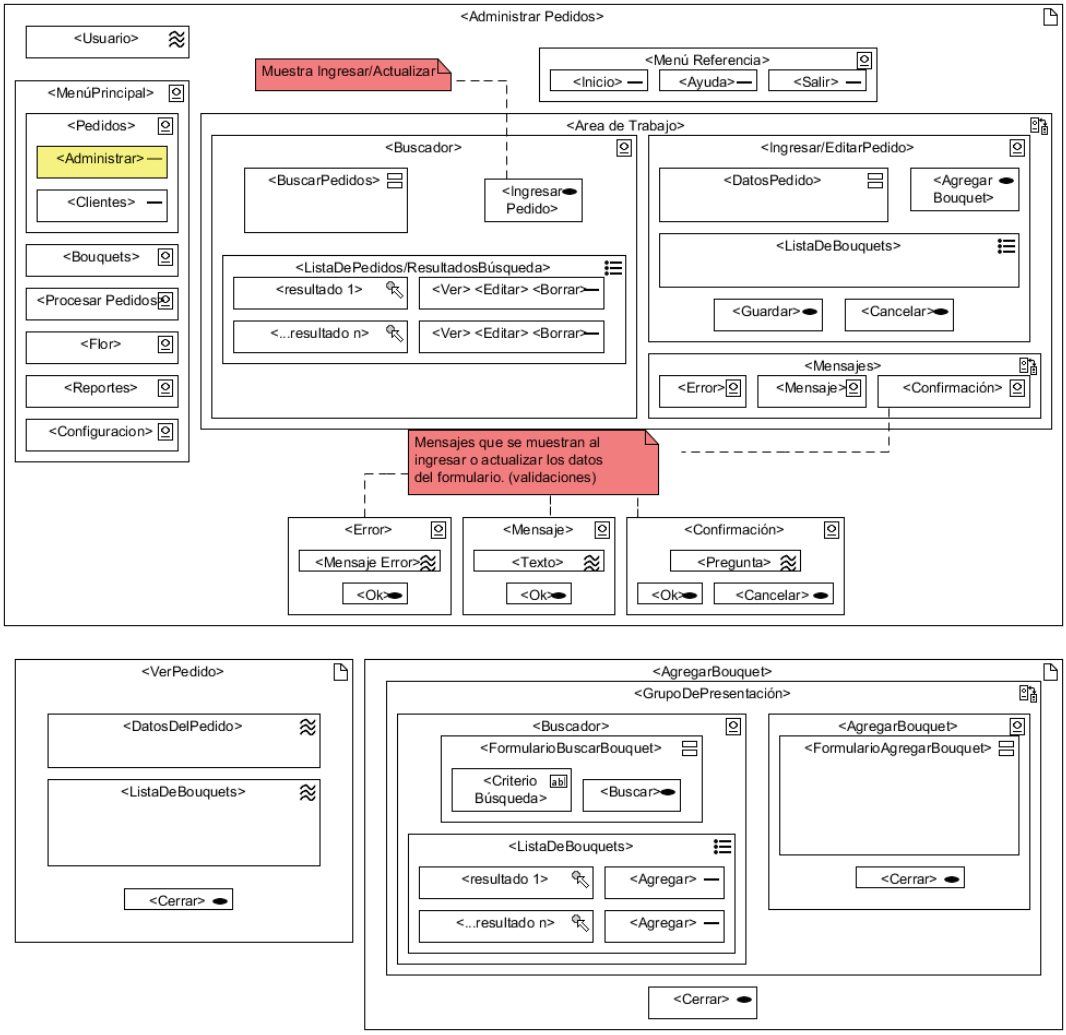
Elaborado por: Carlos Llumiquinga

2.1.5.2 Modelo de presentación Pedidos Bouquets

2.1.5.2.1 Administrar

Se presenta en la Figura 2.21

Figura 2.21: Diagrama de presentación para Pedidos Bouquets



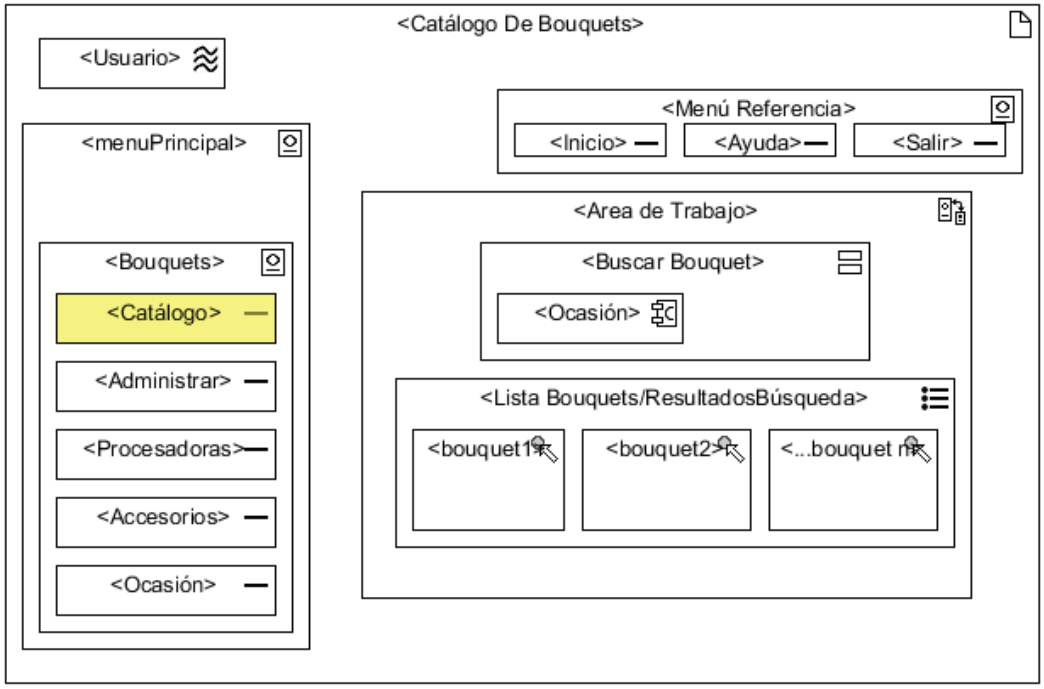
Elaborado por: Carlos Llumiquinga

2.1.5.3 Modelo de presentación Bouquets

2.1.5.3.1 Catálogo

Se presenta en la Figura 2.22

Figura 2.22: Diagrama de presentación para Catálogo de Bouquets

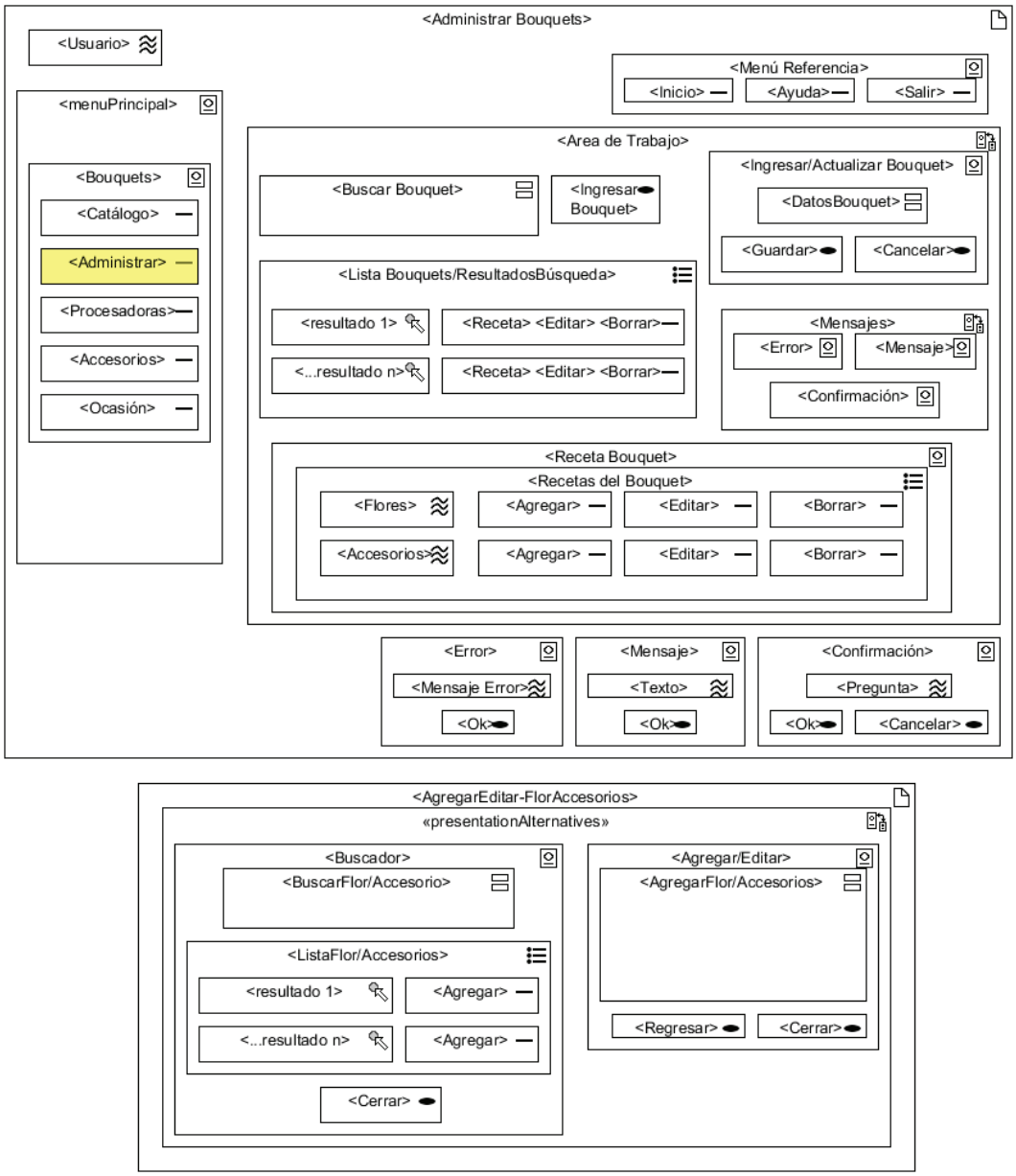


Elaborado por: Carlos Llumiquinga

2.1.5.3.2 Administrar Bouquets

Se presenta en la Figura 2.23

Figura 2.23: Diagrama de presentación para Administrar Bouquets



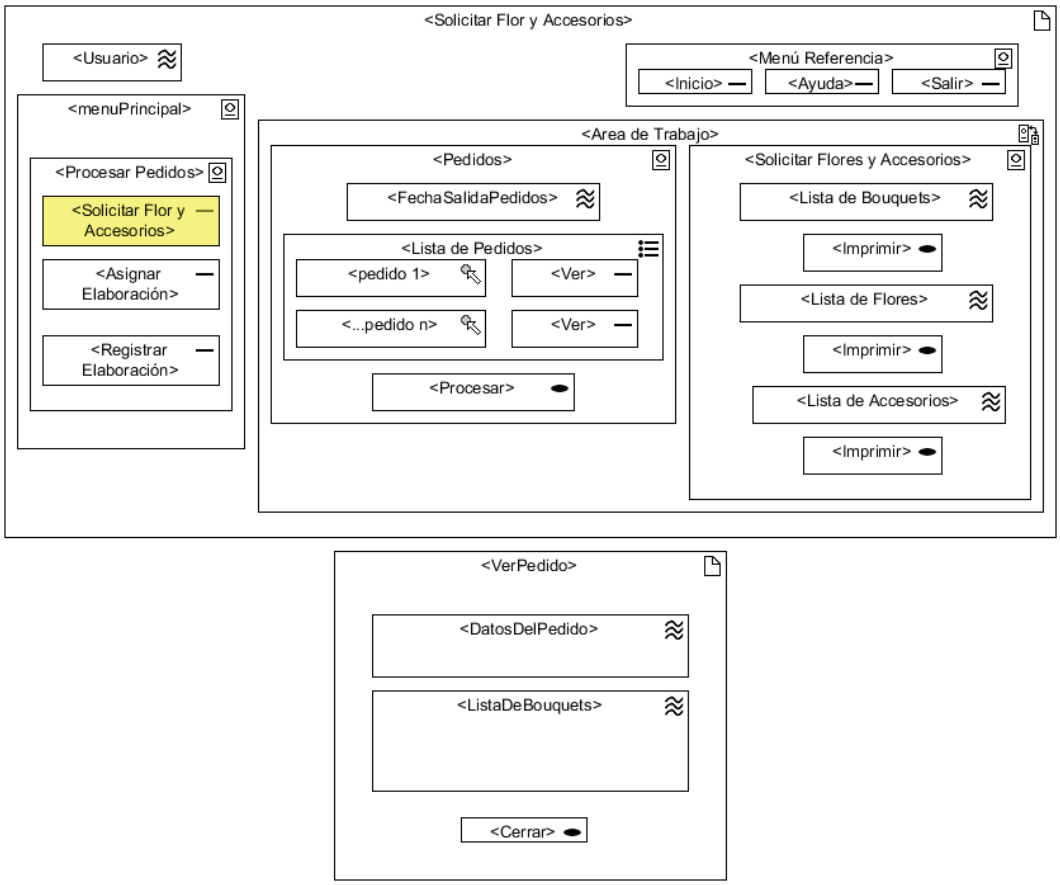
Elaborado por: Carlos Llumiquinga

2.1.5.4 Modelo de presentación Procesar Pedidos

2.1.5.4.1 Solicitar flor y Accesorios

Se presenta en la Figura 2.24

Figura 2.24: Diagrama de presentación para Solicitar Flor y Accesorios

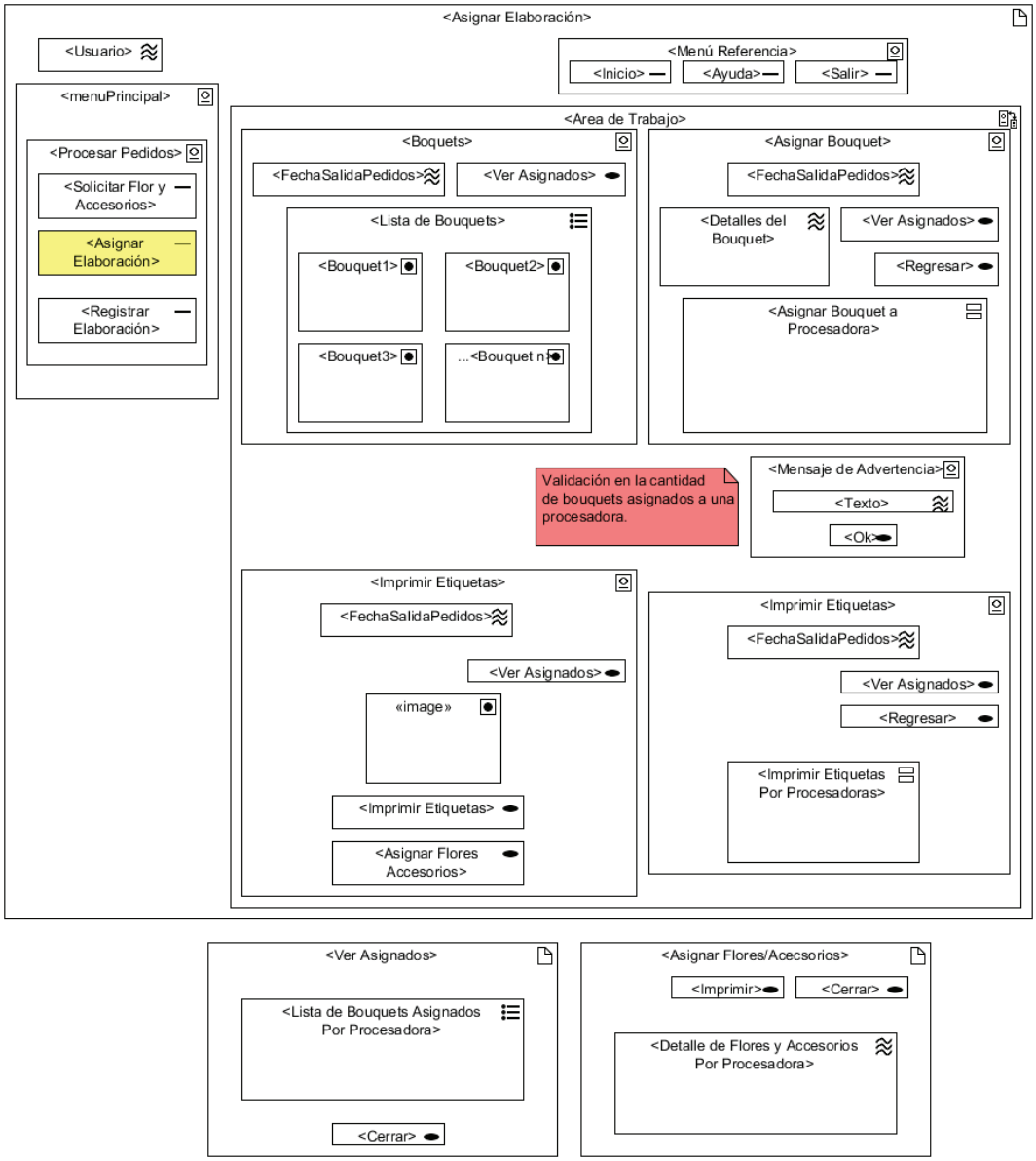


Elaborado por: Carlos Llumiquinga

2.1.5.4.2 Asignar Elaboración

Se presenta en la Figura 2.25

Figura 2.25: Diagrama de presentación para Asignar Elaboración

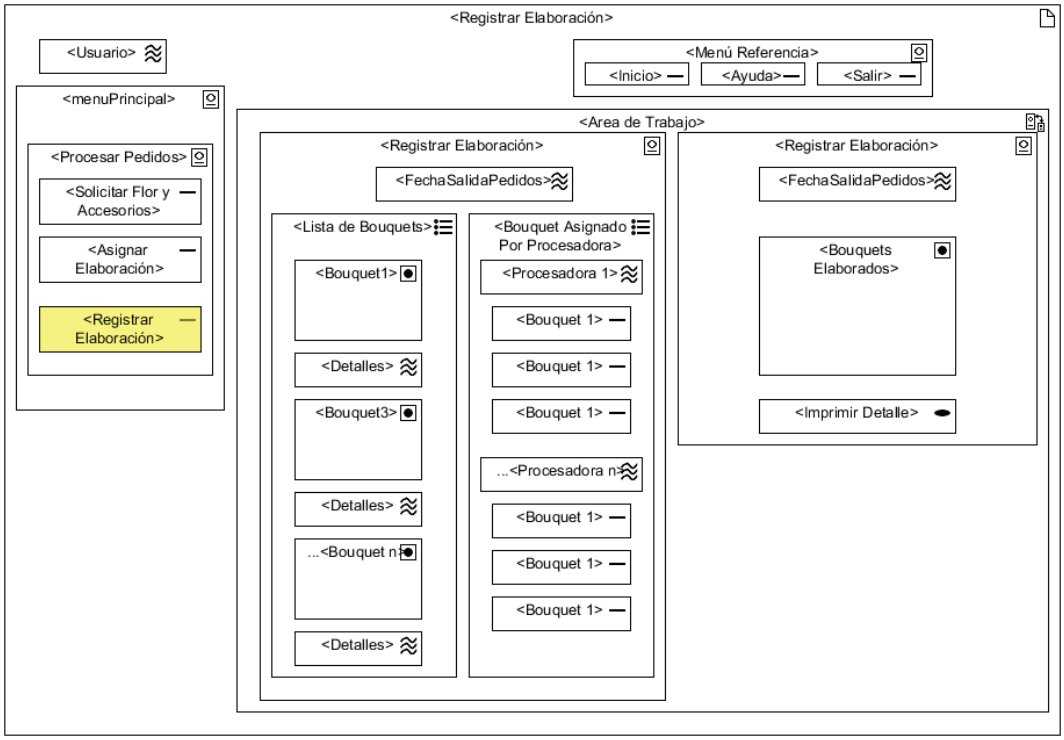


Elaborado por: Carlos Llumiquinga

2.1.5.4.3 Registrar Elaboración

Se presenta en la Figura 2.26

Figura 2.26: Diagrama de presentación para Registrar Elaboración



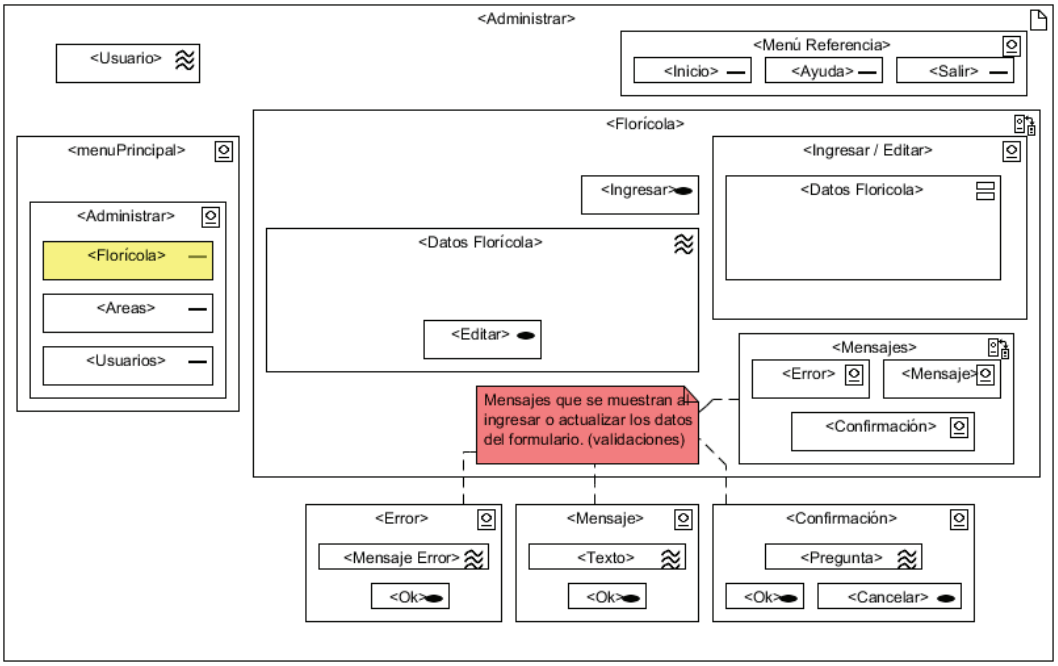
Elaborado por: Carlos Llumiquinga

2.1.5.5 Modelo de presentación Administrar

2.1.5.5.1 Florícola

Se presenta en la Figura 2.27

Figura 2.27: Diagrama de presentación Florícola

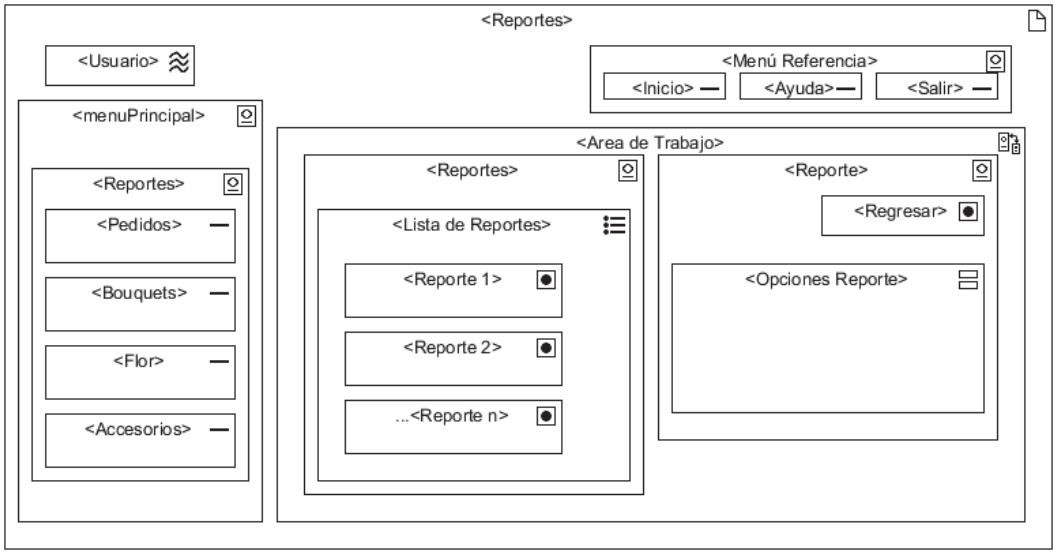


Elaborado por: Carlos Llumiquinga

2.1.5.6 Modelo de presentación Reportes

Se presenta en la Figura 2.28

Figura 2.28: Diagrama de presentación Reportes



Elaborado por: Carlos Llumiquinga

2.2 DISEÑO

2.2.1 ARQUITECTURA DE AKLLASISA

En este punto se mostrará la arquitectura WEB, el patrón de diseño y el modelo de datos utilizados para el desarrollo de la aplicación de acuerdo a los requerimientos establecidos.

2.2.1.1 Arquitectura WEB

Para este proyecto se utilizará una arquitectura de aplicaciones basada en tres capas: presentación, lógica del negocio y acceso a datos, como muestran la Tabla 2.19 y la Figura 2.29.

Tabla 2.19: Capas de la Arquitectura WEB

Capa	Descripción
Presentación	Muestra la aplicación al usuario en forma de una página web, presenta y captura información al usuario la cual es validada controlando que no existan errores. Debe poseer la característica de ser amigable para el usuario.
Lógica de Negocio	Recibe las peticiones del usuario y envía las respuestas luego de ser procesadas. Aquí se establecen un conjunto de pasos y reglas a cumplirse que representan las necesidades que el negocio ha establecido y se le considera la base de la aplicación.
Acceso a Datos	Encargada de acceder a los datos de negocio. Para interactuar con el gestor de base de datos se diseñan operaciones de inserción, consulta, actualización y eliminación de los datos relacionados con el negocio.

Elaborado por: Carlos Llumiquinga

Figura 2.29: Arquitectura WEB

Elaborado por: Carlos Llumiquinga

2.2.1.2 Patrón de diseño

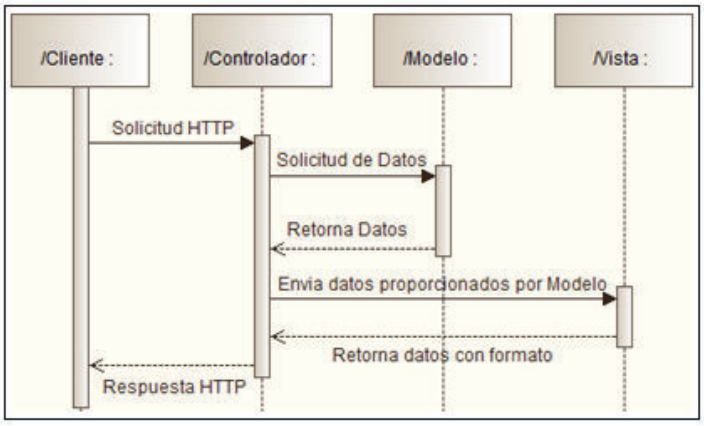
Se utilizará Modelo-Vista-Controlador (MVC), el más usado en desarrollo de aplicaciones web, sobre todo si se utiliza PHP.

MVC separa la aplicación en 3 módulos: modelo, vista y controlador:

- **Modelo**: maneja los datos, los guarda y recupera generalmente de una base de datos, contiene la lógica del negocio implementada por la aplicación.
- **Vista (presentación)**: muestra los datos proporcionados por Modelo en un formato específico, en una aplicación web como una salida HTML.
- **Controlador**: maneja los módulos de Modelo y Vista para que trabajen juntos. Recibe una solicitud del cliente, llama a Modelo para realizar las operaciones solicitadas y envía los datos a Vista.

La Figura 2.30, muestra la secuencia del patrón MVC.

Figura 2.30: Diagrama de Secuencia MVC



Fuente: http://php-html.net/tutorials/model-view-controller-in-php

2.2.1.3 Modelo de Datos

Representado por el Modelo Físico de Datos (Figura 2.31), que se implementará en el gestor de base de datos MySql, seleccionado para alojar la base de datos de la aplicación.

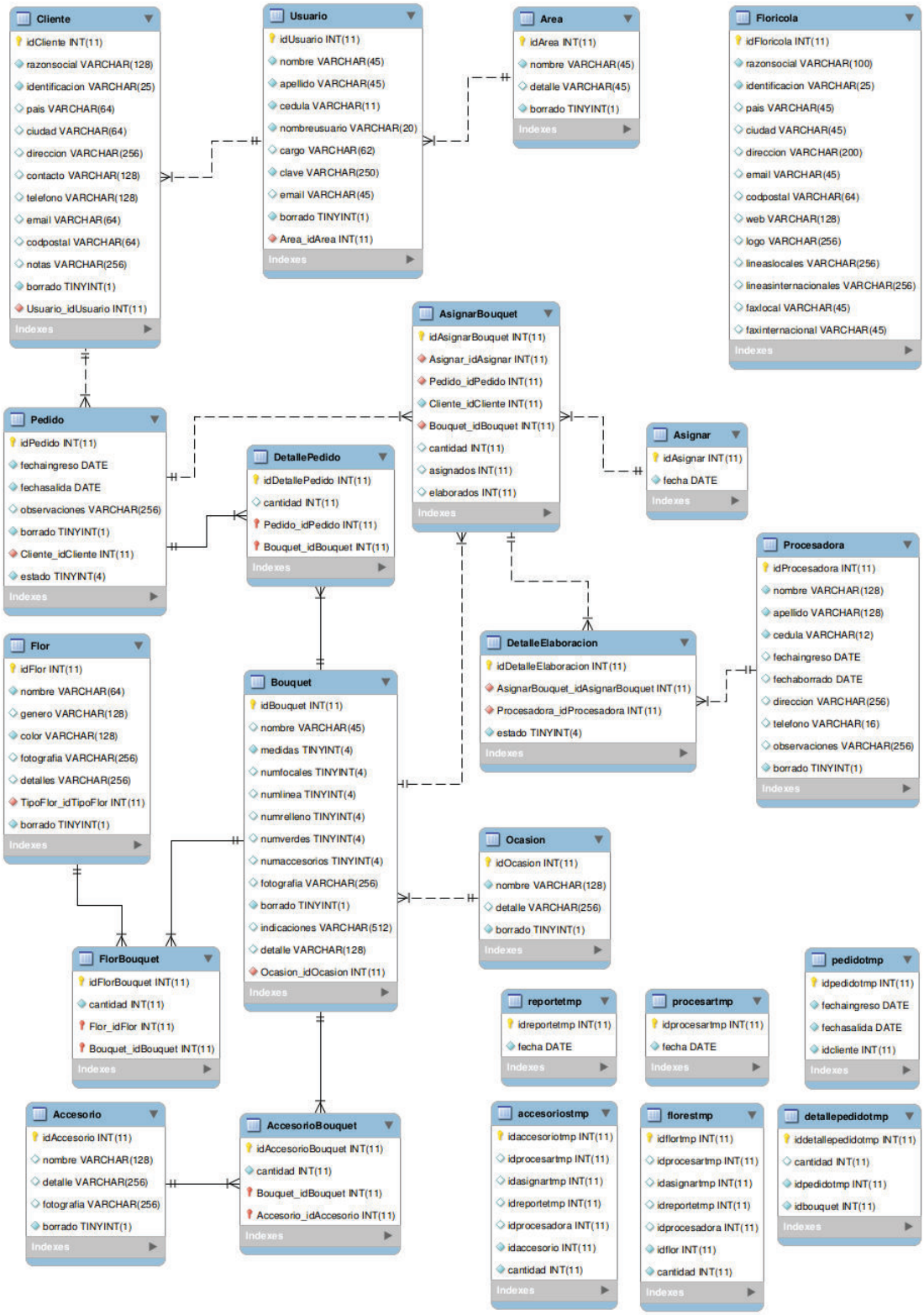
Está basado en el Modelo de Contenido del punto 2.1.3 que es representado por el diagrama UML de clases (Figura 2.13).

El script para la creación de la base de datos se encuentra en la carpeta “Scripts” en el CD adjunto.

Para la construcción del modelo se utilizó la herramienta MySQL Workbench 6.0 Community.

En el Anexo digital C se encuentra el diccionario de datos.

Figura 2.31: Modelo Físico de datos de la aplicación Akllasisa



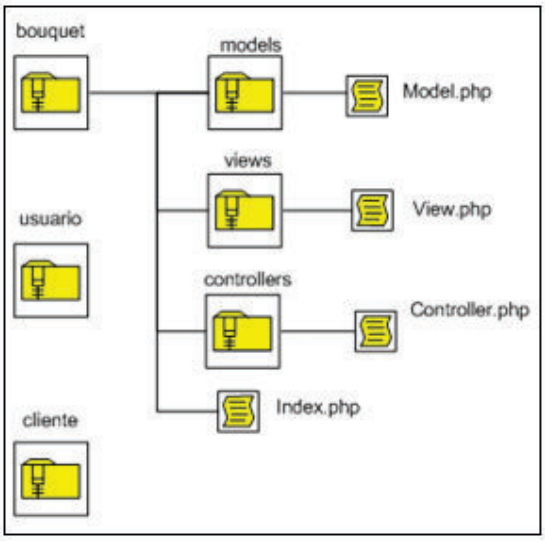
Elaborado por: Carlos Llumiquinga

2.3 CONSTRUCCIÓN DE LA APLICACIÓN

2.3.1 PATRÓN DE DISEÑO MVC APLICADO A AKLLASISA

Para la construcción y desarrollo de Akllasisa, la estructura para codificación tendrá una disposición que coloca los módulos de MVC (Modelo Vista Controlador) en una carpeta disponible para cada caso de uso, se muestra en la Figura 2.32.

Figura 2.32: Patrón de diseño MVC aplicado a Akllasisa



Elaborado por: Carlos Llumiquinga

2.3.2 PROGRAMACIÓN MODULAR

Considerada como un paradigma de la programación, su objetivo es dividir un programa en módulos o subprogramas con el fin de hacerlo más legible y manejable.

Un módulo es cada una de las partes de un programa que se encarga de resolver un subproblema en que se divide a un programa complejo. Cada uno de los cuales tiene una tarea bien definida y pueden necesitar de otros para su operación. Aplicando dicho concepto al desarrollo de la aplicación Akllasisa tendremos que a cada caso de uso lo vamos a considerar como un módulo, los cuales podrán interactuar entre sí para su correcto funcionamiento y tomando en consideración el principio del patrón de diseño MVC visto en el punto anterior.

2.3.3 ESTÁNDAR DE CODIFICACIÓN

Los estándares de código resultan importantes en cualquier proyecto de desarrollo, pero son especialmente importantes cuando muchos desarrolladores trabajan en el mismo proyecto. Los estándares de código ayudan a asegurar que el código tenga una alta calidad, menos errores, y pueda ser mantenido fácilmente. [6]

Para el desarrollo de Akllasisa se va a utilizar los Estándares de Codificación de Zend Framework para PHP http://manual.zfdes.com/es/coding-standard.html.

[6] Estándares de codificación de Zend Framework para PHP, http://manual.zfdes.com/es/coding-standard.html#coding-standard.overview.goals

CAPITULO 3. IMPLEMENTACIÓN Y PRUEBAS

3.1 IMPLEMENTACIÓN

3.1.1 HERRAMIENTAS UTILIZADAS

Para el desarrollo de la aplicación se utilizaron las herramientas mostradas en la Tabla 3.1, cuya selección se realizó en el capítulo 1.3, el entorno para desarrollo y publicación de la aplicación web se utilizará XAMPP que es un servidor independiente de plataforma que contiene MySQL, el Servidor Web Apache y el lenguaje PHP y otras herramientas para administración como phpMyAdmin.

Tabla 3.1: Herramientas Utilizadas

Herramienta	Detalles	Versión
XAMPP: • PHP • MySQL • Apache	 Lenguaje de Programación Base de Datos Servidor Web	1.8.0 5.4.4 5.5.25a 2.4.2
NetBeans PHP	IDE para PHP	7.3.1
Administración Base de Datos	phpMyAdmin	3.5.2

Elaborado por: Carlos Llumiquinga

3.1.2 IMPLEMENTACIÓN DE AKLLASISA

La estructura para la aplicación Akllasisa, de acuerdo al patrón de diseño MVC y aplicando el principio de programación modular que se aplicará en el IDE NetBeans quedará como se muestra en la Figura 3.1:

Figura 3.1: Estructura de Programación modular aplicada a Akllasisa

Elaborado por: Carlos Llumiquinga

- **Application**

  Layouts: contiene los archivos más comunes y usados dentro de la aplicación como son la cabecera, pie de página, entre otros.

  Modules: aquí se encuentra cada módulo que contiene la aplicación.

  - **Module1:** contiene las carpetas que interpretaran al patrón de diseño MVC.

- **Library**
  Contiene todas las librerías que serán usadas por los módulos en la aplicación como son: las necesarias para la conexión a la base de datos, generación de formularios, etc.
- **Public**
  Contiene carpetas de acceso público.
  - **css:** Contiene los archivos de tipo estilos para la aplicación.
  - **js:** Contiene los archivos de tipo java script utilizados en la aplicación.
  - **images:** Contiene todos los archivos de tipo imagen que serán utilizados en la aplicación. Adicionalmente cada módulo contendrá esta carpeta para almacenar imágenes correspondientes al módulo si lo necesita.

3.1.3 CODIFICACIÓN

Ya definidos los estándares para codificación tratados en el capítulo 2.3.3. Se muestra el procedimiento usado para el desarrollo de Akllasisa según MVC.

1) **Conexión a la Base de Datos**

Dentro de la carpeta library tenemos el archivo DbPdo.php que corresponde a una librería de uso público que nos permite realizar la conexión a la base de Datos.

- DbPdo.php
  En la Figura 3.2 se muestra la librería que permite la conexión a la base de datos de la aplicación web.

Figura 3.2: Librería de Conexión a la Base de Datos

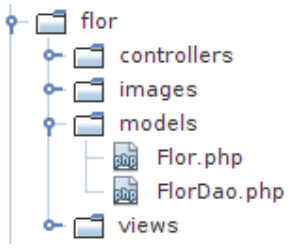
```php
<?php

/**
 * @Descripcion: Libreria de conexion PDO por medio del patron singletom
 * @Adaptada: Jeison Varilla
 * @compartida : Andres Guzman
 * @Licencia: Libre uso GNU-GPL
 * @Email:2012 - keypherc@gamil.com
 * @Website:www.keyphercom.com
 * @Modificaciones: Carlos Llumiquinga
 */

class DbPdo {

    private $_dbh;
    private $_username;
    private $_passwd;
    private $_dns;
    private $_host;
    private $_dbname;

    private static $_instance = null;

    public static function getInstance() {
        if (!(self::$_instance instanceof DbPdo)) {
            self::$_instance = new self();
        }
        return self::$_instance;
    }

    private function __construct() {
        try {
            require_once $_SERVER["DOCUMENT_ROOT"] . '/appAkllasisa/config.ini.php';
            $config = parse_ini_file(BASEPATH . "db.ini", true);
            $this->_username = $config['database']['usuario'];
            $this->_passwd = $config['database']['clave'];
            $this->_host = $config['database']['servidor'];
            $this->_dbname = $config['database']['base'];
            $this->_dns = "mysql:host=" . $this->_host . ";dbname=" . $this->_dbname;
            $this->_dbh = new PDO($this->_dns, $this->_username, $this->_passwd);
            $this->_dbh->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_WARNING);
            $this->_dbh->exec("SET CHARACTER SET utf8");
        } catch (PDOException $e) {
            echo "Error al conectar DB!: " . $e->getMessage() . "<br/>";
            die();
        }
    }

    public function getConn() {
        if ($this->_dbh === null) {
            self::getInstance();
        }

        return $this->_dbh;
    }

    public function isConn() {
        return ((bool) ($this->_dbh instanceof PDO));
    }

    public function closeConn() {
        $this->_dbh = null;
    }

    public function __destruct() {
        $this->closeConn();
    }

}
```

Elaborado por: Carlos Llumiquinga

2) **Crear modelos para las tablas de la Base de Datos**

Dentro de cada módulo el modelo se encarga de la definición y manejo de datos las tablas. Ponemos como ejemplo la tabla flor, que se muestra en las Figuras 3.3 y 3.4:

Figura 3.3: Modelos de la Tabla Flor



Elaborado por: Carlos Llumiquinga

- Flor.php

Figura 3.4: Modelo Flor.php

```php
<?php
/**
 * @Archivo: Flor.php
 * @Modulo: flor
 * @Descripcion: Modelo para tabla Flor
 * @Autor: Carlos Llumiquinga
 *
 */

class Flor {

    private $_idFlor;
    private $_nombre;
    private $_genero;
    private $_color;
    private $_foto;
    private $_detalles;
    private $_idTipoFlor;
    private $_borrado;

    public function get_idFlor() {
        return $this->_idFlor;
    }

    public function set_idFlor($_idFlor) {
        $this->_idFlor = $_idFlor;
    }

    public function get_nombre() {
        return $this->_nombre;
    }

    public function set_nombre($_nombre) {
        $this->_nombre = $_nombre;
    }

```

```php
    public function get_genero() {
        return $this->_genero;
    }

    public function set_genero($_genero) {
        $this->_genero = $_genero;
    }

    public function get_color() {
        return $this->_color;
    }

    public function set_color($_color) {
        $this->_color = $_color;
    }

    public function get_foto() {
        return $this->_foto;
    }

    public function set_foto($_foto) {
        $this->_foto = $_foto;
    }

    public function get_detalles() {
        return $this->_detalles;
    }

    public function set_detalles($_detalles) {
        $this->_detalles = $_detalles;
    }

    public function get_idTipoFlor() {
        return $this->_idTipoFlor;
    }

    public function set_idTipoFlor($_idTipoFlor) {
        $this->_idTipoFlor = $_idTipoFlor;
    }

    public function get_borrado() {
        return $this->_borrado;
    }

    public function set_borrado($_borrado) {
        $this->_borrado = $_borrado;
    }


}
?>
```

Elaborado por: Carlos Llumiquinga

- FlorDao.php: Se muestra la codificación de uno de los métodos para manejar los datos en la tabla. Se muestra en la Figura 3.5.

Figura 3.5: Modelo FlorDao.php

```php
<?php

/**
 * @Archivo: FlorDao.php
 * @Modulo: flor
 * @Descripcion: Modelo, con metodos para manejo de datos de la tabla Flor
 * @Autor: Carlos Llumiquinga
 *
 */
require_once $_SERVER["DOCUMENT_ROOT"] . '/appAkllasisa/config.ini.php';
require_once BASEPATH . "application/modules/flor/models/Flor.php";
require_once BASEPATH . 'library/DbPdo.php';
require_once BASEPATH . 'library/Simple_sessions.php';
require_once BASEPATH . 'library/PDO_Pagination/Pdo_pagination.php';

Class FlorDao {

    /**
     * _getDbh = obtiene el metodo de conexión a la DB mediante singleton
     */
    protected function _getDbh() {
        return DbPdo::getInstance()->getConn();
    }

    /**
     * Ingresa una nueva flor en la tabla
     * @param Flor $flor
     * @return type
     */
    public function ingresar(Flor $flor) {

        //INSERT INTO table (column1, column2) VALUES (value1, value2) WHERE id=key;

        $sql = "INSERT INTO `Flor` (`nombre`,`genero`,`color`, `fotografia`, `detalles`, `TipoFlor_
                . "VALUES(?, ?, ?, ?, ?, ?)";

        $stm = $this->_getDbh()->prepare($sql);
        $stm->bindvalue(1, $flor->get_nombre(), PDO::PARAM_STR);
        $stm->bindvalue(2, $flor->get_genero(), PDO::PARAM_STR);
        $stm->bindvalue(3, $flor->get_color(), PDO::PARAM_STR);
        $stm->bindvalue(4, $flor->get_foto(), PDO::PARAM_STR);
        $stm->bindvalue(5, $flor->get_detalles(), PDO::PARAM_STR);
        $stm->bindvalue(6, $flor->get_idTipoFlor(), PDO::PARAM_STR);

        try {
            $result = $stm->execute();
            //echo $this->_getDbh()->lastInsertId();
            //exit;
            return $result;
        } catch (PDOException $e) {
            echo $e->getMessage();
        }
    }
```

Elaborado por: Carlos Llumiquinga

3) **Crear Vistas**

Tomando el mismo ejemplo de la Tabla Flor, se muestra un ejemplo de una vista para este módulo.

- verFlor.php

```
<?php

/**
 * @Archivo: verFlor.php
 * @Modulo: flor
 * @Descripcion: Muestra todos los datos de una Flor
 * @Autor: Carlos Llumiquinga
 *
 */

require_once $_SERVER["DOCUMENT_ROOT"] . '/appAkllasisa/config.ini.php';
include_once BASEPATH . 'application/layouts/sesion.php';
require_once BASEPATH . 'library/zebra_form/Zebra_Form.php';


if ($_SESSION['idarea'] == '1' || $_SESSION['idarea'] == '5') {
    ?>

    <link type="text/css" rel="stylesheet" href="<?php echo BASEURL; ?>public/css/pagina.css" />
    <link type="text/css" rel="stylesheet" href="<?php echo BASEURL; ?>public/css/estilos.css" />
    <link type="text/css" rel="stylesheet" href="<?php echo BASEURL;
?>library/zebra_form/public/css/zebra_form.css"/>
    <link type="text/css" rel="stylesheet" href=" <?php echo BASEURL . "public/css/tablaDatos.css" ?>" />
    <script type="text/javascript" src="<?php echo BASEURL; ?>public/js/jquery-1.9.1.min.js"></script>

    <script type="text/javascript" src="<?php echo BASEURL;
?>library/zebra_form/public/javascript/zebra_form.js"></script>
    <meta charset="UTF-8">

    <script language="javascript">

        var cursor;
        if (document.all) {
            // Está utilizando EXPLORER
            cursor = 'hand';
        } else {
            // Está utilizando MOZILLA/NETSCAPE
            cursor = 'pointer';
        }
    </script>
    <title>Ver Flor</title>

    <?php
    if (isset($_GET['id']) && !empty($_GET['id'])) {
        $idf = $_GET['id'];
        require_once BASEPATH . 'application/modules/flor/models/FlorDao.php';
        $obj_f = new FlorDao();
        $flor = $obj_f->buscarFlorID($idf);
        ?>

        <div align="center">
            <h3>VER FLOR</h3><br>
            <table width="80%" id="tablaDatos">
                <tr><th colspan="2" >
                        <?php echo $flor[0]['nombre'] ?>
                    </th>
                </tr>
                <tr class=alt>
                    <td style="font-weight: bold;" width="20%">Género:</td><td><?php echo $flor[0]['genero']
?></td>
                </tr>
                <tr class=alt>
                    <td style="font-weight: bold;" width="20%">Color:</td><td><?php echo $flor[0]['color'] ?></td>
```

```
            </tr>
            <tr class=alt>
               <td style="font-weight: bold;" width="20%">Tipo Flor:</td><td><?php echo $flor[0]['nombretf']
?></td>
            </tr>
            <tr class=alt>
               <td style="font-weight: bold;" width="20%">Detalle:</td><td><?php echo $flor[0]['detalles']
?></td>
            </tr>
            <tr >
               <td colspan="2" align="center">
                  <img src="<?php echo BASEURLFLOR . $flor[0]['fotografia'] ?>" width="240" height="177" />
               </td>
            </tr>
         </table>
      </div>
      <br/>
      <div align="center">
         <table>
            <tr>
               <td>
                  <img alt="Cerrar" title="Cerrar" src="<?php echo BASEURL
?>public/images/botones/cerrar_16.png" onClick="window.close()" border="1" onMouseOver="style.cursor
= cursor">
               </td>
            </tr>
         </table>
      </div>
      <?php
   }
}
?>
```

Donde se puede apreciar la estructura de la codificación, usando los archivos de configuración, conexión a la base de datos, librerías y los modelos creados en los ejemplos anteriores. Además del respectivo código html para la presentación de los datos.

4) **Crear Controladores**

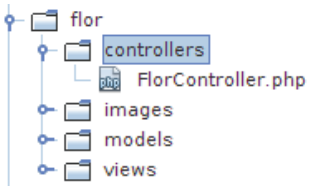
Encargados de la interacción entre las vistas y los modelos de la aplicación, su interacción se muestra en la implementación de formularios.

En este punto vamos a revisar el controlador del módulo flor en el proceso de ingresar una nueva flor en la tabla Flor, donde el formulario para ingresar la nueva flor envía los datos al Controlador el cual interactúa con los modelos y las vistas del módulo.

- FlorController.php

El ejemplo se encuentra descrito en las Figuras 3.6 y 3.7.

Figura 3.6: Controlador Módulo Flor



Elaborado por: Carlos Llumiquinga

Figura 3.7: Controlador FlorController.php

```php
<?php

/**
 * @Archivo: FlorController.php
 * @Modulo: flor
 * @Descripcion: Controlador para módulo Flor
 * @Autor: Carlos Llumiquinga
 *
 */
require_once $_SERVER["DOCUMENT_ROOT"] . '/appAkllasisa/config.ini.php';
require_once BASEPATH . 'library/Inputfilter.php';
require_once BASEPATH . 'library/Helpers.php';
require_once BASEPATH . 'application/modules/flor/models/Flor.php';
require_once BASEPATH . 'application/modules/flor/models/FlorDao.php';


if (isset($_GET['insertar']) && $_GET['insertar'] == 'Insertar') {
    // Validación para que las variables no esten vacias
    if (!Helpers::validDatas(array($_GET['nombre'], $_GET['genero'], $_GET['color']))) {
        // Error! Existen datos vacios
        header('location:' . BASEURL . 'application/modules/flor/views/insFlor.php?m=1');
    } else {
        $obj_clean = new InputFilter();
        $nombre = $obj_clean->process($_GET['nombre']);
        $genero = $obj_clean->process($_GET['genero']);
        $color = $obj_clean->process($_GET['color']);
        $foto = $obj_clean->process($_GET['foto']);
        $detalle = $obj_clean->process($_GET['detalle']);
        $idTipoFlor = $obj_clean->process($_GET['selTF']);

        $obj_flor = new FlorDao();
        //Validación para que el nombre de la Flor sea único en la tabla
        $response = $obj_flor->checkNombreFlor($nombre);

        if ($response['total'] >= 1) {
            // Ya existe una Flor con ese nombre
```

Vista

```
37              header('location:' . BASEURL . 'application/modules/flor/views/insFlor.php?m=3');
38          } elseif ($response['total'] == 0) {
39              $nuevaFlor = new Flor();
40              $nuevaFlor->set_nombre($nombre);
41              $nuevaFlor->set_genero($genero);
42              $nuevaFlor->set_color($color);
43              $nuevaFlor->set_foto($foto);
44              $nuevaFlor->set_detalles($detalle);
45              $nuevaFlor->set_idTipoFlor($idTipoFlor);
46              if (!empty($foto)) {
47                  $nuevaFlor->set_foto($foto);
48              } else {
49                  $nuevaFlor->set_foto('noDisponible.png');
50              }
51              // Método para ingresar nueva Flor
52              $result = $obj_flor->ingresar($nuevaFlor);
53              if ($result == null) {
54                  //No se pudo guardar
55                  header('location:' . BASEURL . 'application/modules/flor/views/index.php?m=2');
56              } else {
57                  //Se ha guardado en la DB
58                  header('location:' . BASEURL . 'application/modules/flor/views/index.php?m=4');
59              }
60          }
61      }
62  }
```

Vista

Modelo Flor

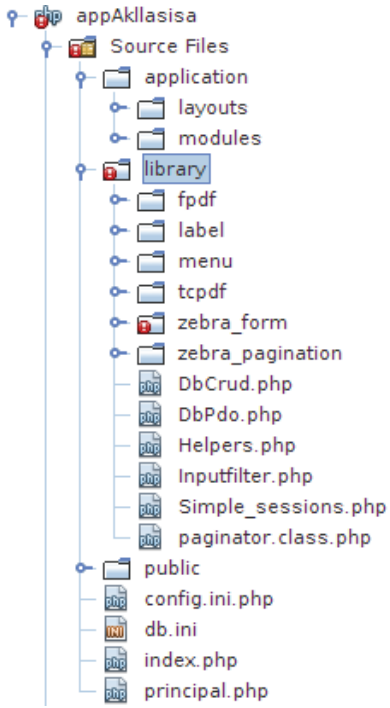
Modelo FlorDao

Elaborado por: Carlos Llumiquinga

5) **Librerías**

La carpeta library contiene todas las librerías que se usaron para el desarrollo de Akllasisa. Se muestra en la Figura 3.8.

Figura 3.8: Librerías



Elaborado por: Carlos Llumiquinga

El código fuente de toda la aplicación se encuentra dentro de la Carpeta “Código Fuente” del CD adjunto.

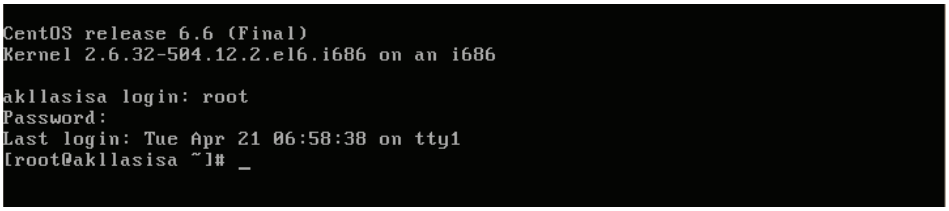
3.1.4 INSTALACIÓN DE AKLLASISA

Para un correcto funcionamiento de Akllasisa se provee de todo el entorno necesario y configurado, por medio de XAMPP en donde encuentra la aplicación, permitiendo así un correcto funcionamiento sin la necesidad de librerías extras o configuraciones adicionales.

3.1.4.1 Proceso de Instalación

1) Ingresar a la terminal de comandos en el servidor con las credenciales del usuario root.

Figura 3.9: Terminal de Comandos



2) Colocar el CD en la unidad de CD/DVD Rom del servidor.

3) Montar el CD en CentOS, seguir los siguientes pasos:
Ejecutar el comando:

mount /dev/dvd /media

4) Ejecutar el instalador, digitando el siguiente comando en la terminal:

/media/INSTALADOR/akllasisa.run

Figura 3.10: Ejecutar el Instalador

5) El instalador comienza el proceso:

a) Creará el usuario akllasisa.

Nombre de Usuario: akllasisa

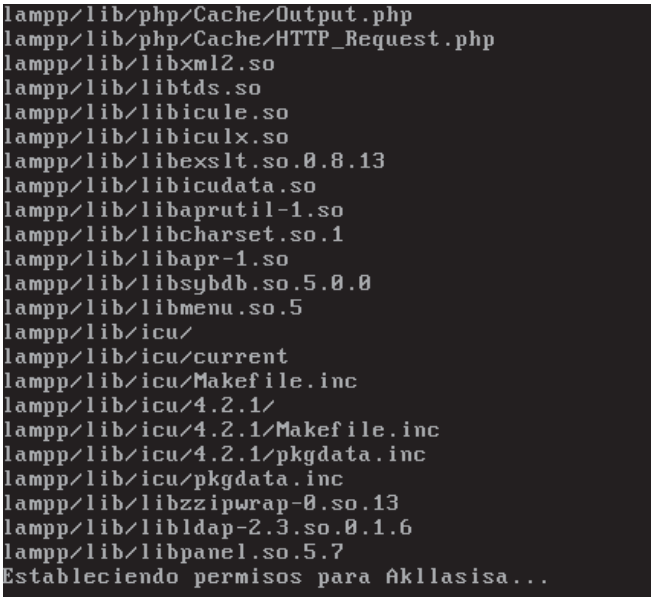
Clave: bouquets2015

Figura 3.11: Usuario de Akllasisa

b) Extrae los archivos necesarios.

c) Establece los permisos necesarios.

Figura 3.12: Extracción y permisos de Akllasisa

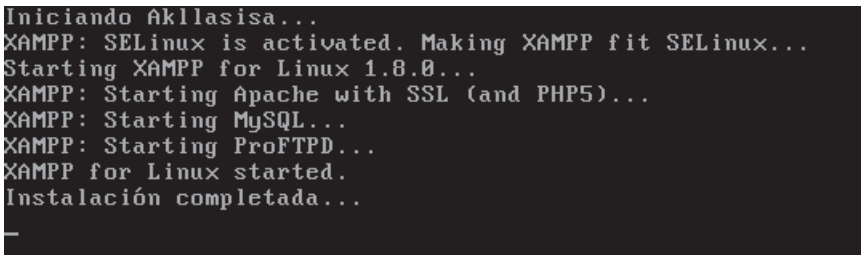


d) Solicita el ingreso de la Dirección IP del servidor, si se ignora se usará el valor: **localhost**.

Figura 3.13: Ingresar IP del Servidor

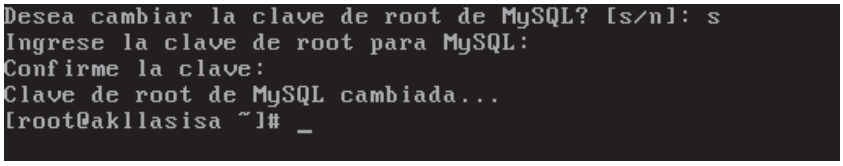
e) Inicia XAMPP

Figura 3.14: Inicio XAMPP



f) Por seguridad, se consulta si se desea realizar el cambio de la clave del usuario root de MySQL. Si se ignora queda la clave por defecto: **12345678**

Figura 3.15: Clave de root para MySQL



3.1.4.2 Base de Datos

La estructura completa de la Base de Datos para el funcionamiento de la aplicación está incluida dentro de XAMPP revisado en el punto anterior.

En las tablas 3.2, 3.3 y 3.4 los datos iniciales de algunas de las tablas más importantes de Akllasisa.

3.1.4.2.1 Datos Iniciales Tabla Área

Tabla 3.2: Datos Iniciales Tabla Área

ID	Nombre	Detalle
1	Sistemas	Área de Sistemas Informáticos
2	Gerencia	Área de Gerentes
3	Ventas	Área de Ejecutivos de Ventas
4	Bouquetera	Área de Bouquetera
5	Postcosecha	Área de Postcosecha

Elaborado por: Carlos Llumiquinga

3.1.4.2.2 Datos Iniciales Tabla Usuario

Tabla 3.3: Datos Iniciales Tabla Usuario

ID	Nombre	Apellido	NombreUsuario	Clave	idArea
1	Administrador	Akllasisa	admin	123456	1
2	Gerente	Florícola	gerente	123456	2
3	Ventas	Florícola	ventas	123456	3
4	Bouquetera	Florícola	bouquetera	123456	4
5	Postcosecha	Florícola	postcosecha	123456	5

Elaborado por: Carlos Llumiquinga

3.1.4.2.3 Datos Iniciales Tabla Tipo Flor

Tabla 3.4: Datos Iniciales Tabla Tipo Flor

ID	Nombre	Detalle
1	Flor de Relleno	Racimos de flores individuales en un solo tallo
2	Flor Verde	Complementan el aspecto del bouquet
3	Flor Focal	Flores llamativas del bouquet
4	Flor de Línea	Establecen la altura y anchura del bouquet

Elaborado por: Carlos Llumiquinga

3.1.4.3 Archivos de Configuración de Akllasisa

Akllasisa necesita la configuración de 2 archivos:

a) config.ini.php: Archivo de configuración para constantes globales requeridas por la aplicación, se muestra en la Figura 3.16.

Figura 3.16: Archivo de configuración config.ini.php

```php
<?php
/**
 * @Descripcion: Config.ini.php
 *  Contiene constantes globales que se requieren en la
 *  mayoría de las operaciones de base de datos o archivo
 * @author: Carlos Llumiquinga - 2015
 */

// Definición de constantes para el manejo de URL absolutas
    define('BASEURL','http://192.168.1.5/appAkllasisa/');
    define('BASEPATH',dirname(__FILE__).'/');
    define('BASEURLACCESORIO',BASEURL.'application/modules/accesorio/images'.'/');
    define('BASEPATHACCESORIO',BASEPATH.'application/modules/accesorio/images'.'/');
    define('BASEURLFLOR',BASEURL.'application/modules/flor/images'.'/');
    define('BASEPATHFLOR',BASEPATH.'application/modules/flor/images'.'/');
    define('BASEPATHDATOS',BASEPATH.'application/modules/floricola/images'.'/');
    define('BASEURLDATOS',BASEURL.'application/modules/floricola/images'.'/');
    define('BASEPATHBOU',BASEPATH.'application/modules/bouquet/images'.'/');
    define('BASEURLBOU',BASEURL.'application/modules/bouquet/images'.'/');

?>
```

Elaborado por: Carlos Llumiquinga

b) db.ini: Contiene los parámetros necesarios para la conexión a la Base de Datos. Se muestran en la Figura 3.17.

Figura 3.17: Archivo de configuración db.ini

```ini
; Parámetros para Conexión a la base de datos
[database]
usuario = "root"
clave = "12345678"
servidor = "localhost"
base = "dbBouquets"
```

Elaborado por: Carlos Llumiquinga

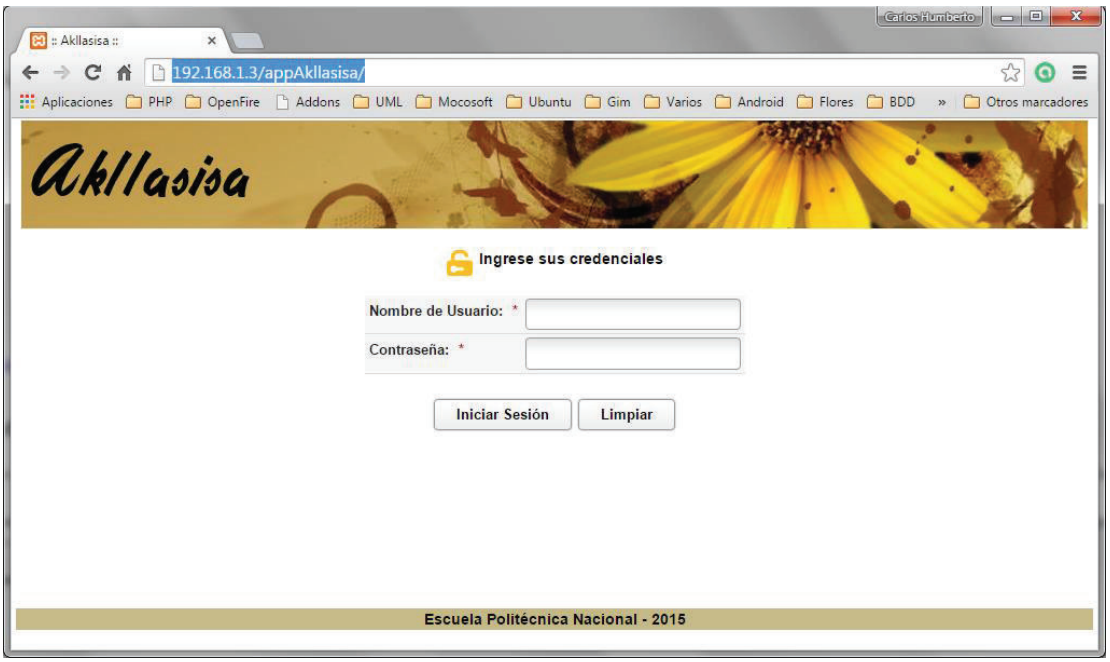
3.1.4.4 Ingresar a la Aplicación

Dependiendo de la dirección IP que fue asignada al servidor, para este caso: 192.168.1.3, en un navegador de internet desde un computador que se encuentre en la red de la florícola ingresar a la siguiente dirección:

http://192.168.1.3/appAkllasisa

La página de ingreso de la aplicación se muestra en la Figuras 3.18. Luego de ingresar las credenciales aparece la página principal que muestra la Figura 3.19.

Figura 3.18: Página de Ingreso



Elaborado por: Carlos Llumiquinga

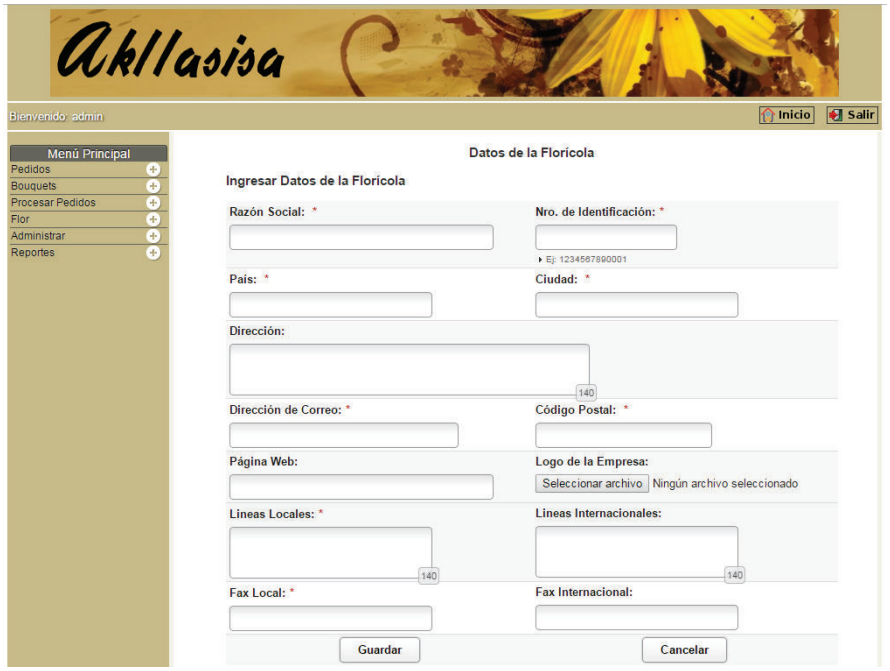
Figura 3.19: Página Principal

Elaborado por: Carlos Llumiquinga

3.1.4.5 Datos de la Florícola

Este paso es de importancia ya que los datos de la florícola son necesarios para el ingreso de los Pedidos por parte de los ejecutivos de ventas, como se muestra en la Figura 3.20.

Figura 3.20: Ingreso Datos de la Florícola



Elaborado por: Carlos Llumiquinga

3.2 PRUEBAS

Con el objetivo de asegurar que la aplicación fue desarrollada según los requerimientos, se realizaron las siguientes pruebas a los casos de uso más representativos.

3.2.1 EVALUACIÓN CASO DE ESTUDIO

3.2.1.1 Caso de Estudio BellaFlor-Group

Las casos de pruebas fueron realizadas con la ayuda de empleados que pertenecen a cada área de la florícola BellaFlor-Group.

Como se menciona anteriormente se tomarán los casos de uso más representativos para comprobar el funcionamiento de la aplicación.

3.2.1.1.1 Caso de Prueba Administrar Usuarios

Tabla 3.5: Prueba Ingresar Usuario

Usuario	Administrador
Caso de Uso	Ingresar Usuario
Entradas	**Nombre:** Juan Carlos **Apellido:** Dávila **Cédula de Identidad:** 1734253469 **Nombre de Usuario:** jdavila **Clave:** 12345678 **Área:** Ventas **Correo Electrónico:** jdavila@bellaflor-group.com **Teléfono:** 2345667
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que debe ser ingresado. - Mensaje de Advertencia. - Mensaje de usuario guardado.
Descripción	El Administrador ingresa un nuevo usuario.
Procedimiento de Prueba	1. En el Menú Administrar, seleccionar la opción Usuarios. 2. Seleccionar Ingresar Usuario. 3. Ingresar los datos del Usuario. 4. Clic en el botón guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación número de cédula. - Validación nombre de usuario sea único.

Elaborado por: Carlos Llumiquinga

Tabla 3.6: Prueba Buscar Usuario

Usuario	Administrador
Caso de Uso	Buscar Usuario
Entradas	**Buscar (Nombre):** Juan Carlos
Salidas	- Datos de Usuarios cuyo nombre se Juan Carlos.
Descripción	El Administrador busca un usuario.
Procedimiento de Prueba	1. En el Menú Administrar, seleccionar la opción Usuarios. 2. Ingresar y seleccionar el parámetro de búsqueda. 3. Clic en el botón Buscar.
Resultados Esperados	Akllasisa busca el usuario según los parámetros de búsqueda en la Base de Datos y los muestra.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Si no se ingresa y se selecciona el parámetro de búsqueda se muestran todos los usuarios existentes.

Elaborado por: Carlos Llumiquinga

Tabla 3.7: Prueba Actualizar Usuario

Usuario	Administrador
Caso de Uso	Actualizar Usuario
Entradas	**Nombre:** Juan Carlos **Apellido:** Dávila Jurado **Cédula de Identidad:** 1734253469 **Nombre de Usuario:** jdavila **Clave:** 12345678 **Área:** Ventas **Correo Electrónico:** jdavila@floricola.com **Teléfono:** 2345667
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que deben ser ingresados.

	- Mensaje de Advertencia. - Mensaje de usuario actualizado.
Descripción	El Administrador edita los datos de un Usuario.
Procedimiento de Prueba	1. En el Menú Administrar, seleccionar la opción Usuarios. 2. Buscar el Usuario. 3. Seleccionar Editar Usuario. 4. Modificar los datos del Usuario. 5. Clic en el botón Guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación número de cédula. - Validación nombre de usuario sea único.

Elaborado por: Carlos Llumiquinga

Tabla 3.8: Prueba Eliminar Usuario

Usuario	Administrador
Caso de Uso	Eliminar Usuario
Entradas	Ninguna
Salidas	- Mensaje de Advertencia. - Mensaje de usuario eliminado
Descripción	El Administrador elimina un Usuario.
Procedimiento de Prueba	1. En el Menú Administrar, seleccionar la opción Usuarios. 2. Busca el Usuario. 3. Seleccionar Borrar Usuario. 4. Confirmar Borrar el Usuario.
Resultados Esperados	- Mensaje “Usuario Eliminado”. - Akllasisa elimina el Usuario de la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- No se puede eliminar el usuario Administrador.

Elaborado por: Carlos Llumiquinga

3.2.1.1.2 Caso de Prueba Administrar Clientes

Tabla 3.9: Prueba Ingresar Cliente

Usuario	Ejecutivo de Ventas
Caso de Uso	Ingresar Cliente
Entradas	**Razón Social:** Florería El Lirio **Num. Identificación:** 1784569879001 **País:** Ecuador **Ciudad:** Quito **Dirección:** Av. 6 de Diciembre y Eloy Alfaro **Nombre del Contacto:** Juan Salas **Num. Teléfono:** 2568741 **Dirección de Correo:** ellirio@gmail.com **Código Postal:** EC170122 **Notas:** Ninguna **Ejecutivo de Ventas:** Ventas Florícola
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que deben ser ingresados. - Mensaje de Advertencia. - Mensaje de Cliente Ingresa con éxito.
Descripción	El Ejecutivo de Ventas ingresa un nuevo cliente.
Procedimiento de Prueba	1. En el Menú Pedidos, seleccionar la opción Clientes. 2. Seleccionar Agregar. 3. Ingresar los datos del Cliente. 4. Clic en el botón guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación número de identificación. - Validación de razón social y número de identificación sean únicos.

Elaborado por: Carlos Llumiquinga

Tabla 3.10: Prueba Buscar Cliente

Usuario	Ejecutivo de Ventas
Caso de Uso	Buscar Cliente
Entradas	**Buscar (Razón Social):** Florería El Lirio
Salidas	- Datos del Cliente cuya razón social sea “Florería el Lirio”
Descripción	El Ejecutivo de Ventas busca un cliente.
Procedimiento de Prueba	1. En el Menú Pedidos, seleccionar la opción Clientes. 2. Ingresar y seleccionar el parámetro de búsqueda. 3. Clic en el botón Buscar.
Resultados Esperados	Akllasisa busca el cliente según los parámetros de búsqueda en la Base de Datos y los muestra.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Si no se ingresa y se selecciona el parámetro de búsqueda se muestran todos los clientes existentes del Ejecutivo de Ventas.

Elaborado por: Carlos Llumiquinga

Tabla 3.11: Prueba Editar Cliente

Usuario	Ejecutivo de Ventas
Caso de Uso	Editar Cliente
Entradas	**Razón Social:** Florería El Lirio **Num. Identificación:** 1784569879001 **País:** Ecuador **Ciudad:** Quito **Dirección:** Av. 6 de Diciembre y Eloy Alfaro **Nombre del Contacto:** Juan Salas **Num. Teléfono:** 2568741 **Dirección de Correo:** ellirio@gmail.com **Código Postal:** EC170122 **Notas:** Se debe llamar los días martes

	Ejecutivo de Ventas: Ventas Florícola
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que deben ser ingresados. - Mensaje de Advertencia. - Mensaje de Cliente Ingresa con éxito.
Descripción	El Ejecutivo de Ventas ingresa un nuevo cliente.
Procedimiento de Prueba	1. En el Menú Pedidos, seleccionar la opción Clientes. 2. Buscar el Cliente 3. Seleccionar Editar Cliente. 4. Modificar los datos del Cliente. 5. Clic en el botón guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación número de identificación. - Validación de razón social y número de identificación sean únicos.

Elaborado por: Carlos Llumiquinga

Tabla 3.12: Prueba Eliminar Cliente

Usuario	Ejecutivo de Ventas
Caso de Uso	Eliminar Cliente
Entradas	**Ninguna**
Salidas	- Mensaje de Advertencia. - Mensaje de Cliente eliminado
Descripción	El Ejecutivo de Ventas elimina un cliente.
Procedimiento de Prueba	1. En el Menú Pedidos, seleccionar la opción Clientes. 2. Busca el Cliente. 3. Seleccionar Borrar Cliente. 4. Confirmar Borrar el Cliente.

Resultados Esperados	- Mensaje “Cliente Eliminado”. - Akllasisa elimina el Cliente de la Base de datos
Resultados Obtenidos	Satisfactorios
Casos de Excepción	Ninguno

Elaborado por: Carlos Llumiquinga

3.2.1.1.3 Caso de Prueba Administrar Flor

Tabla 3.13: Prueba Ingresar Flor

Usuario	Postcosecha
Caso de Uso	Ingresar Flor
Entradas	**Nombre:** Black Bacara **Género:** Rosa **Color:** Rojo **Detalle:** Botón de la flor normal, arreglos serios **Tipo de Flor:** Flor Focal **Fotografía:** Archivo blackbacara.jpg.
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que debe ser ingresado. - Mensaje de Advertencia. - Mensaje de Flor guardada.
Descripción	El Usuario Postcosecha ingresa una nueva flor.
Procedimiento de Prueba	1. En el Menú Flor, seleccionar la opción Administrar. 2. Seleccionar Agregar. 3. Ingresar los datos de la Flor. 4. Clic en el botón guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación nombre de la Flor sea único.

	- Validación formato y tamaño de la fotografía de la Flor.

Elaborado por: Carlos Llumiquinga

Tabla 3.14: Prueba Buscar Flor

Usuario	Postcosecha
Caso de Uso	Buscar Flor
Entradas	**Buscar (Nombre):** Black
Salidas	- Datos de Flor cuyo nombre contenga Black.
Descripción	El Usuario Postcosecha busca una Flor.
Procedimiento de Prueba	1. En el Menú Flor, seleccionar la opción Administrar. 2. Ingresar y seleccionar el parámetro de búsqueda. 3. Clic en el botón Buscar.
Resultados Esperados	Akllasisa busca la Flor según los parámetros de búsqueda en la Base de Datos y los muestra.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Si no se ingresa y se selecciona el parámetro de búsqueda se muestran todas las Flores existentes.

Elaborado por: Carlos Llumiquinga

Tabla 3.15: Prueba Actualizar Flor

Usuario	Postcosecha
Caso de Uso	Actualizar Flor
Entradas	**Nombre:** Black Bacara **Género:** Rosa **Color:** Rojo sangre **Detalle:** Botón de la flor normal, arreglos serios **Tipo de Flor:** Flor Focal

Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que deben ser ingresados. - Mensaje de Advertencia. - Mensaje de Flor actualizada.
Descripción	El Usuario Postcosecha edita los datos de una Flor.
Procedimiento de Prueba	1. En el Menú Flor, seleccionar la opción Administrar. 2. Buscar la Flor. 3. Seleccionar Editar Flor. 4. Modificar los datos de la Flor. 5. Clic en el botón Guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación nombre de la Flor sea único. - Validación formato y tamaño de la fotografía de la Flor.

Elaborado por: Carlos Llumiquinga

Tabla 3.16: Prueba Eliminar Flor

Usuario	Postcosecha
Caso de Uso	Eliminar Flor
Entradas	Ninguna
Salidas	- Mensaje de Advertencia. - Mensaje de Flor eliminada.
Descripción	El Usuario Postcosecha elimina una Flor.
Procedimiento de Prueba	1. En el Menú Flor, seleccionar la opción Administrar. 2. Buscar la Flor. 3. Seleccionar Borrar Flor. 4. Confirmar Borrar la Flor.

Resultados Esperados	- Mensaje “Flor Eliminada”. - Akllasisa elimina la Flor de la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- No se puede eliminar la Flor si consta en la receta de un Bouquet.

Elaborado por: Carlos Llumiquinga

3.2.1.1.4 Caso de Prueba Administrar Bouquets

Tabla 3.17: Prueba Ingresar Bouquet

Usuario	Bouquetera
Caso de Uso	Ingresar Bouquet
Entradas	**Nombre:** Sweet Love **Medida(cm): 50** **Num. Focales:** 6 **Num. Línea:** 5 **Num. Relleno:** 0 **Num. Verdes:** 4 **Num. Accesorios:** 1 **Ocasión:** San Valentín **Detalle:** Arreglo con predominio de color Rojo **Fotografía:** Archivo sweetlove.jpg.
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que debe ser ingresado. - Mensaje de Advertencia. - Mensaje de Bouquet guardado. - Ingresar Receta del Bouquet
Descripción	El Usuario Bouquetera ingresa un nuevo bouquet.
Procedimiento de Prueba	1. En el Menú Bouquet, seleccionar la opción Administrar. 2. Seleccionar Agregar. 3. Ingresar los datos del Bouquet.

	4. Clic en el botón guardar.
Resultados Esperados	- Akllasisa valida la información ingresada y la registra en la Base de datos. - Se muestra la página para ingresar la receta del Bouquet.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación que el nombre del Bouquet sea único. - Validación del formato y tamaño de la fotografía del Bouquet.

Elaborado por: Carlos Llumiquinga

Tabla 3.18: Prueba Ingresar/Modificar Receta

Usuario	Bouquetera
Caso de Uso	Ingresar/Modificar Receta
Entradas	**Flores Focales** **Nombre:** Black Beauty **Cantidad: 6** **Flores de Línea** **Nombre:** Snap Dragon Hot Pink **Cantidad: 5** **Flores de Relleno** Ninguna **Flores Verdes** **Nombre:** Greens **Cantidad:** 4 **Accesorios** **Nombre:** Papel Celofán Rojo **Cantidad: 1** **Indicaciones:** Revisar la fotografía
Salidas	- Mensaje de Advertencia, si se supera la cantidad de flores de la receta genérica. - Flor ingresada en la receta. - Imprimir receta del Bouquet.
Descripción	El Usuario Bouquetera ingresa la receta detallada de un bouquet.
Procedimiento de Prueba	1. En el Menú Bouquet, seleccionar la opción Administrar.

	2. Seleccionar Ver Receta. 3. Ingresar los datos de la receta del Bouquet.
Resultados Esperados	- Akllasisa valida la información ingresada y la registra en la Base de datos. - Imprimir la Receta del Bouquet.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación cantidad del tipo de Flor no exceda al de la receta genérica.

Elaborado por: Carlos Llumiquinga

Tabla 3.19: Prueba Buscar Bouquet

Usuario	Bouquetera
Caso de Uso	Buscar Bouquet
Entradas	**Buscar (Nombre):** Sweet
Salidas	- Datos de Bouquet cuyo nombre contenga Sweet.
Descripción	El Usuario Bouquetera busca un Bouquet.
Procedimiento de Prueba	1. En el Menú Bouquet, seleccionar la opción Administrar. 2. Ingresar y seleccionar el parámetro de búsqueda. 3. Clic en el botón Buscar.
Resultados Esperados	Akllasisa busca los Bouquets según los parámetros de búsqueda en la Base de Datos y los muestra.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Si no se ingresa y se selecciona el parámetro de búsqueda se muestran todas los Bouquets existentes.

Elaborado por: Carlos Llumiquinga

Tabla 3.20: Prueba Actualizar Bouquet

Usuario	Bouquetera
Caso de Uso	Actualizar Bouquet
Entradas	**Nombre:** Sweet Love **Medida(cm): 50** **Num. Focales:** 6 **Num. Línea:** 5 **Num. Relleno:** 0 **Num. Verdes:** 4 **Num. Accesorios:** 1 **Ocasión:** San Valentín **Detalle:** Arreglo con predominio de color Rojo y contraste con verde
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que deben ser ingresados. - Mensaje de Advertencia. - Mensaje de Bouquet actualizado.
Descripción	El Usuario Bouquetera edita los datos de un Bouquet.
Procedimiento de Prueba	1. En el Menú Bouquets, seleccionar la opción Administrar. 2. Buscar el Bouquet. 3. Seleccionar Editar Bouquet. 4. Modificar los datos del Bouquet. 5. Clic en el botón Guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación nombre del Bouquet sea único.

Elaborado por: Carlos Llumiquinga

Tabla 3.21: Prueba Eliminar Bouquet

Usuario	Bouquetera
Caso de Uso	Eliminar Bouquet
Entradas	Ninguna
Salidas	- Mensaje de Advertencia. - Mensaje de Bouquet eliminado.
Descripción	El Usuario Bouquetera elimina un Bouquet.
Procedimiento de Prueba	1. En el Menú Bouquets, seleccionar la opción Administrar. 2. Buscar el Bouquet. 3. Seleccionar Borrar Bouquet. 4. Confirmar Borrar el Bouquet.
Resultados Esperados	- Mensaje “Bouquet Eliminado”. - Akllasisa elimina lógicamente el Bouquet de la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	Ninguno

Elaborado por: Carlos Llumiquinga

3.2.1.1.5 Caso de Prueba Administrar Pedidos

Tabla 3.22: Prueba Ingresar Pedido

Usuario	Ejecutivo de Ventas
Caso de Uso	Ingresar Pedido
Entradas	**Fecha de Ingreso:** 24/02/2015 **Fecha de Salida:** 26/02/2015 **Cliente:** Florería el Lirio **Bouquet(s):** Sweet Love **Cantidad:** 10
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que debe ser ingresado. - Mensaje de Advertencia. - Mensaje de Pedido guardado.

Descripción	El Usuario Ventas ingresa un nuevo Pedido.
Procedimiento de Prueba	1. En el Menú Pedidos, seleccionar la opción Administrar. 2. Seleccionar Ingresar Pedido. 3. Ingresar los datos del Pedido. 4. Clic en el botón guardar.
Resultados Esperados	- Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación que se haya seleccionado un cliente. - Validación ingresados ítems al pedido no se puede cambiar el cliente del pedido. - Validación fecha de salida del pedido.

Elaborado por: Carlos Llumiquinga

Tabla 3.23: Prueba Buscar Pedido

Usuario	Ejecutivo de Ventas
Caso de Uso	Buscar Pedido
Entradas	**Buscar (Razón Social):** Lirio
Salidas	- Datos del Pedido donde la Razón social del Cliente contenga Lirio.
Descripción	El Ejecutivo de Ventas busca un Pedido.
Procedimiento de Prueba	1. En el Menú Bouquet, seleccionar la opción Administrar. 2. Ingresar y seleccionar el parámetro de búsqueda. 3. Clic en el botón Buscar.
Resultados Esperados	Akllasisa busca los Pedidos según los parámetros de búsqueda en la Base de Datos y los muestra.
Resultados Obtenidos	Satisfactorios

Casos de Excepción	- Si no se ingresa y se selecciona el parámetro de búsqueda se muestran todas los Bouquets existentes.

Elaborado por: Carlos Llumiquinga

Tabla 3.24: Prueba Actualizar Pedido

Usuario	Ejecutivo de Ventas
Caso de Uso	Actualizar Pedido
Entradas	**Fecha de Ingreso:** 24/02/2015 **Fecha de Salida:** 26/02/2015 **Cliente:** Florería el Lirio **Bouquet(s):** Sweet Love **Cantidad:** 5
Salidas	- Los campos con * son obligatorios, sino se ingresan se muestra un mensaje de que deben ser ingresados. - Mensaje de Advertencia. - Mensaje de Pedido actualizado.
Descripción	El Ejecutivo de Ventas edita los datos de un Pedido.
Procedimiento de Prueba	1. En el Menú Pedidos, seleccionar la opción Administrar. 2. Buscar el Pedido. 3. Seleccionar Editar Pedido. 4. Modificar los datos del Pedido. 5. Clic en el botón Guardar.
Resultados Esperados	Akllasisa valida la información ingresada y la registra en la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación si el estado del pedido es “Ingresado” se permite editar el pedido. - Validación Fecha de Salida no haya expirado.

Elaborado por: Carlos Llumiquinga

Tabla 3.25: Prueba Cancelar Pedido

Usuario	Ejecutivo de Ventas
Caso de Uso	Cancelar Pedido
Entradas	Ninguna
Salidas	- Mensaje de Advertencia. - Mensaje de Bouquet eliminado.
Descripción	El Ejecutivo de Ventas elimina un Pedido.
Procedimiento de Prueba	1. En el Menú Pedidos, seleccionar la opción Administrar. 2. Buscar el Pedido. 3. Seleccionar Cancelar Pedido. 4. Confirmar Cancelar el Pedido.
Resultados Esperados	- Mensaje “Pedido Cancelado”. - Akllasisa elimina lógicamente el Pedido de la Base de datos.
Resultados Obtenidos	Satisfactorios
Casos de Excepción	- Validación si el estado del pedido es “Ingresado” se permite Cancelar el pedido. - Validación Fecha de Salida no haya expirado.

Elaborado por: Carlos Llumiquinga

3.3 EVALUACIÓN DE RESULTADOS

A continuación se presentan los resultados obtenidos de las pruebas realizadas a la aplicación web Akllasisa, aplicadas al caso de estudio la florícola BellaFlor-Group, con la ayuda de sus empleados.

Las pruebas realizadas con ingreso de información, no revelaron inconsistencia alguna además del correcto funcionamiento de los controles de validación de la información que se ingresa.

Las pruebas de los Casos de Uso más representativos, muestran el correcto funcionamiento de la aplicación en las actividades cotidianas involucradas en la elaboración de un Bouquet que realiza la florícola.

Los procesos se ejecutaron en tiempos aceptables dentro de la red de la Florícola.

El ingreso a la aplicación solo se permite con las respectivas credenciales de cada usuario y el acceso a cada opción es controlado por el área al que pertenece el usuario, los intentos por ingresar a la aplicación sin las credenciales respectivas y a las opciones que no le corresponde a su área fueron sin éxito.

3.3.1 PRUEBAS DE ACEPTACIÓN

A los empleados que participaron en las pruebas se les realizó un cuestionario, cuyo formato se encuentra en el Anexo digital D.

El cuestionario se realizó a un total de 12 empleados que pertenecen a las áreas de la florícola que manejaron la aplicación, se muestran en la Tabla 3.26.

Tabla 3.26: Muestra de Empleados que respondieron el Cuestionario

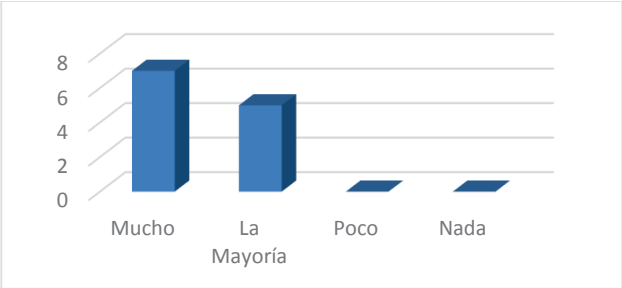
Área	Número de Empleados
Sistemas	1
Ventas	5
Bouquetera	2
PostCosecha	2
Gerencia	2
Total:	**12**

Elaborado por: Carlos Llumiquinga

A continuación se presentan los resultados que arrojó el cuestionario realizado:

- Los resultados mostrados en la figura 3.21 corresponden a la Pregunta 1: ¿Forma en que la aplicación ayuda a sus tareas diarias?

Figura 3.21: Resultados Pregunta 1 del Cuestionario

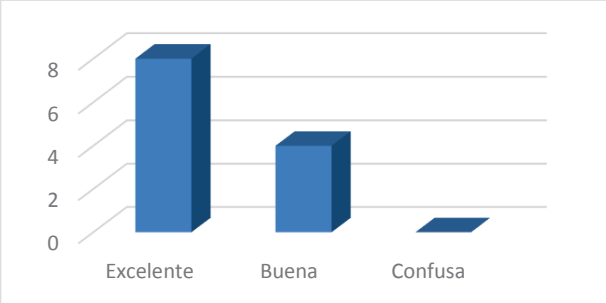


Elaborado por: Carlos Llumiquinga

Los usuarios consideran que la aplicación es de gran ayuda en sus actividades diarias.

- Los resultados mostrados en la figura 3.22 corresponden a la Pregunta 2: ¿Cómo le resulta la navegación a través de la aplicación?

Figura 3.22: Resultados Pregunta 2 del Cuestionario

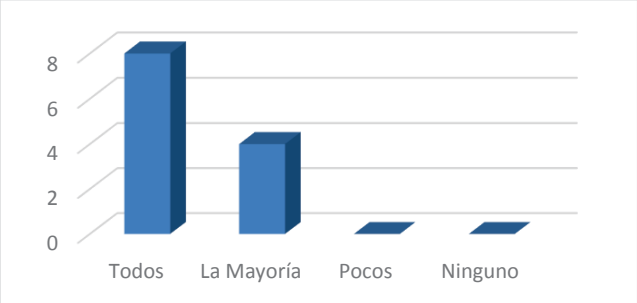


Elaborado por: Carlos Llumiquinga

Los usuarios consideran que la navegación por el sistema no presenta dificultad.

- Los resultados mostrados en la figura 3.23 corresponden a la Pregunta 3: ¿Los iconos de la aplicación representan la función de los mismos?

Figura 3.23: Resultados Pregunta 3 del Cuestionario

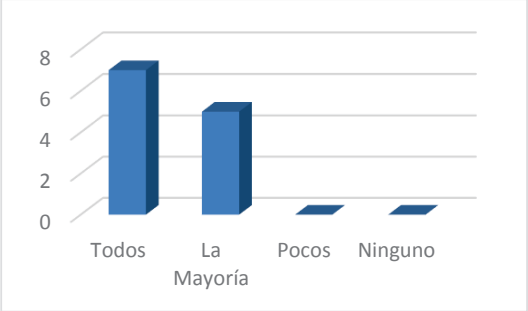


Elaborado por: Carlos Llumiquinga

Los usuarios consideran que los iconos en la aplicación representan con claridad la función que tienen asignada.

- Los resultados mostrados en la figura 3.24 corresponden a la Pregunta 4: ¿Son claros los nombres y descripciones que aparecen en la aplicación?

Figura 3.24: Resultados Pregunta 4 del Cuestionario

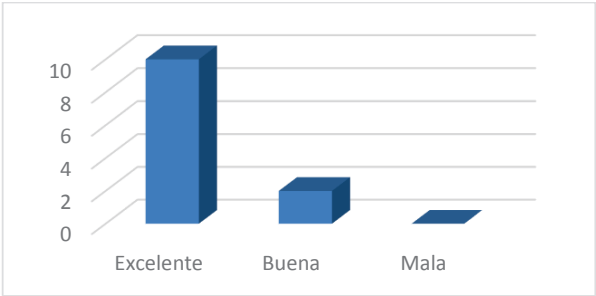


Elaborado por: Carlos Llumiquinga

Los usuarios consideran que la mayoría de los nombres y descripciones dentro de la aplicación son claros y no presentan dificultad para comprenderlos.

- Los resultados mostrados en la figura 3.25 corresponden a la Pregunta 5: La legibilidad del tipo y tamaño de letra de la aplicación es.

Figura 3.25: Resultados Pregunta 5 del Cuestionario

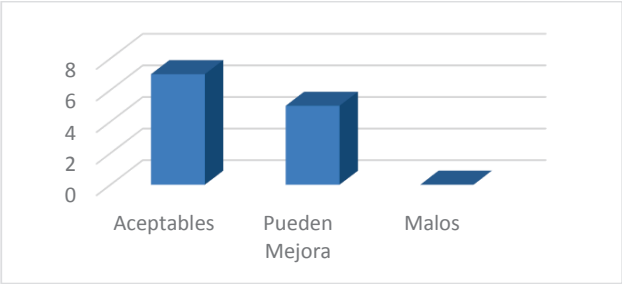


Elaborado por: Carlos Llumiquinga

Los usuarios consideran que el tipo de letra y tamaño usado es legible.

- Los resultados mostrados en la figura 3.26 corresponden a la Pregunta 6: ¿Cómo califica los colores y otros elementos empleados en la aplicación?

Figura 3.26: Resultados Pregunta 6 del Cuestionario

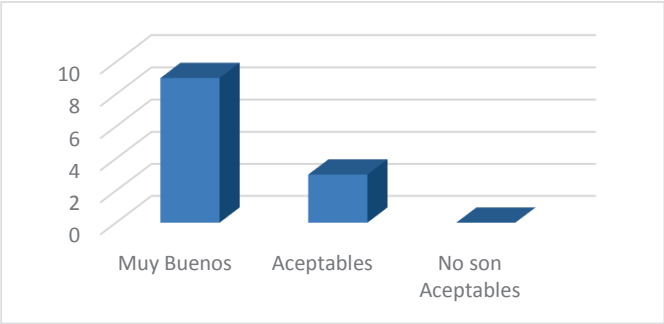


Elaborado por: Carlos Llumiquinga

Los colores y otros elementos en su mayoría son considerados aceptables para los usuarios.

- Los resultados mostrados en la figura 3.27 corresponden a la Pregunta 7: ¿Cómo le parecieron los tiempos de respuesta de la aplicación?

Figura 3.27: Resultados Pregunta 7 del Cuestionario

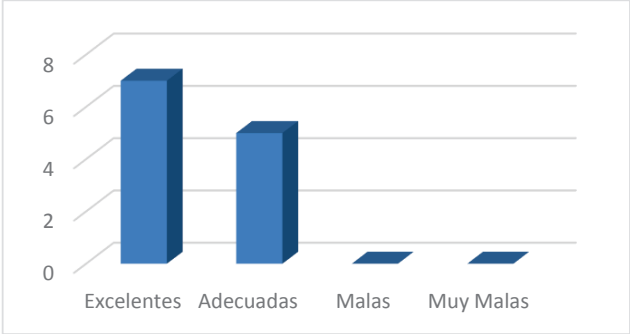


Elaborado por: Carlos Llumiquinga

En general los usuarios consideran que los tiempos de respuesta de la aplicación como buenos.

- Los resultados mostrados en la figura 3.28 corresponden a la Pregunta 8: La estructura de las secciones mostradas en la página son:

Figura 3.28: Resultados Pregunta 8 del Cuestionario

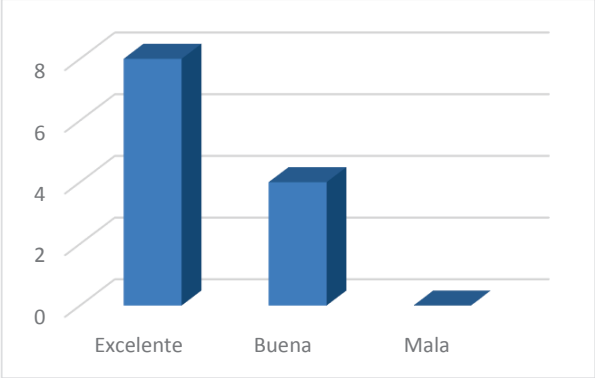


Elaborado por: Carlos Llumiquinga

Los usuarios consideran que la estructura de las páginas de la aplicación en su mayoría es adecuada.

- Los resultados mostrados en la figura 3.29 corresponden a la Pregunta 9: ¿Cuál es su calificación de la apariencia general de la aplicación?

Figura 3.29: Resultados Pregunta 9 del Cuestionario

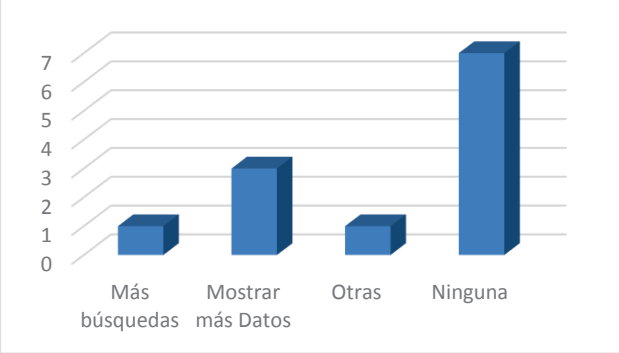


Elaborado por: Carlos Llumiquinga

La apariencia de la aplicación es considerada en su mayoría como excelente para los usuarios.

- Los resultados mostrados en la figura 3.30 corresponden a la Pregunta 10: ¿Qué opciones le hacen falta a la aplicación?

Figura 3.30: Resultados Pregunta 10 del Cuestionario



Elaborado por: Carlos Llumiquinga

Los usuarios consideran que debe existir más presentación de información y búsqueda, mientras para otros las opciones existentes son suficientes.

Los cuestionarios llenados por los empleados se encuentran en el Anexo digital E.

CAPITULO 4. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

El uso de UWE, permite mostrar que es una extensión de UML y el Proceso Unificado de Software muy poderosa para el desarrollo de aplicaciones WEB, logrando representar las vistas de la aplicación web en este proyecto, dando como resultado los diagramas de navegación y presentación, que representan la estructura de navegación y el diseño de la interfaz del usuario respectivamente.

La metodología usada combinada con UWE permitió definir de manera precisa los requerimientos del usuario y desarrollar un diseño consistente de la aplicación, permitiendo adaptar nuevos requerimientos que aparecieron durante el proceso de desarrollo.

La aplicación WEB ha permitido interactuar a todos los procesos involucrados en la elaboración de un Bouquet, desde que el Ejecutivo de Ventas ingresa un pedido de bouquets y este comienza su proceso para que los bouquets sean elaborados y despachados de la Bouquetera, logrando un proceso más eficiente y centralizado.

La información proporcionada por la aplicación permite obtener datos que son aprovechados por las gerencias de una florícola con el fin de establecer parámetros de rendimiento, uso y administración de los recursos involucrados.

Todas las herramientas para el Desarrollo de este proyecto son basadas en Software de Libre Difusión, mostrando la variedad y robustez que existe frente a las herramientas privativas, de igual manera lo fácil que resulta armar un entorno de desarrollo teniendo a disposición librerías, herramientas de modelado, IDEs entre otros, sin costo alguno.

4.2 RECOMENDACIONES

Para una correcta definición de los requerimientos es recomendable interactuar en los procesos involucrados y con sus actores, ya sea por medio de entrevistas y/o participando de sus actividades diarias.

Se debe verificar la consistencia de los diagramas de navegación, con el fin de proporcionar al usuario una fácil navegación a través de la aplicación Web.

Una capacitación respecto a las herramientas usadas para el correcto funcionamiento y administración de la aplicación web, es necesaria al personal de sistemas o tecnología.

Respecto a la integridad de la base de datos, una política de respaldos adecuada la garantiza ante cualquier falla que se presente.

Se recomienda agregar el manejo de precios de los bouquets en la aplicación en una futura versión.

La impresión de etiquetas en formatos A4 usando impresoras de tipo normal, en algunas ocasiones representa desperdicio de las mismas. Para una futura versión se recomienda implementar la impresión de las etiquetas por medio de impresoras tipo zebra.

Akllasisa no se enfoca solamente al uso dentro de una florícola, se recomienda su uso para negocios como florerías que pueden necesitar un registro de recetas de sus arreglos florales y entre otras opciones que presenta la aplicación.

REFERENCIAS BIBLIOGRÁFICAS

Libros:

- RUMBAUGH, James; JACOBSON, Ivar; BOOCH, Grady. “The Unified Modeling Language Reference Manual Second Edition”. Addison-Wesley Pearson Education, Inc. 2005.
- RUMBAUGH, James; JACOBSON, Ivar; BOOCH, Grady. “El Proceso Unificado de Desarrollo de Software”. Addison-Wesley Pearson Education, Inc. 2000.
- FOWLER, Martin. “UML Distilled: A Brief Guide to the Standard Object Modeling Language”. Addison-Wesley Professional; 3rd edition, September 25, 2003.
- ROSSI, Gustavo; SCHWAVE, Daniel; PASTOR, Oscar; OLSINA, Luis. “Web Engineering: Modelling and Implementing Web Applications”. Springer-Verlag London Limited 2008.
- NIXON, Robin. “PHP, MySQL, JavaScript & CSS”. O’Reilly Media 2nd Edition, August 2012.
- KOFLER, Michael. “The Definitive Guide to MySQL 5”. Apress 3rd Edition 2005

Direcciones Electrónicas

- UWE – UML-based Web Engineering. http://uwe.pst.ifi.lmu.de/ Último Acceso 2015-02-03.
- KEYPHERM. http://www.keyphercom.com/ Último acceso 2014-11-11.
- MySQL Documentation http://dev.mysql.com/doc/ Último Acceso 2014-08-12.
- Mini referencia de SQL con MySQL. http://crysol.github.io/recipe/2006-05-16/mini-referencia-de-sql-con-mysql/#.VO7HNXyG_K8. Último Acceso 2014-11-20.
- Stefan Gabos, Zebra Forms http://stefangabos.ro/wp-content/docs/Zebra_Form/li_Controls.html Último acceso 2015-01-18.
- Blog sobre UWE (UML-based Web Engineering) http://uwespanishblog.blogspot.com Último Acceso, 2014-12-09

ANEXOS DIGITALES

Se encuentran dentro de la carpeta ANEXOS en el CD adjunto.

Anexo A

Contiene los acrónimos y abreviaturas que se usaron.

Anexo B

Contiene las reglas del negocio

Anexo C

Contiene el diccionario de datos del modelo físico de la base de datos.

Anexo D

Contiene el formato del cuestionario realizado para las pruebas de aceptación.

Anexo E

Contiene los cuestionarios llenados por los empleados para las pruebas de aceptación.