

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**

**FACULTAD DE INGENIERÍA**

**ESCUELA DE SISTEMAS**



**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SISTEMAS Y COMPUTACIÓN**

**“DESARROLLO DE UN SISTEMA WEB PARA LA ADMINISTRACIÓN Y  
PRESENTACIÓN DE INFORMACIÓN DE LA BASE DE DATOS DEL HERBARIO  
QCA DE LA PUCE”**

**AUTORES:**

**DANIEL ALEJANDRO ANGULO GRANDA**

**DIEGO GERMAN FREIRE FIGUEROA**

**DIRECTOR**

**MSC. DAMIÁN ANIBAL NICOLALDE RODRÍGUEZ**

**QUITO, MARZO 2017**

## Contenidos

Índice de Tablas .....	8
Índice de Ilustraciones .....	9
Índice de Diagramas .....	10
Prefacio .....	11
Antecedentes.....	12
Justificación .....	13
Objetivo General.....	14
Objetivos Específicos .....	14
Alcance.....	14
Capítulo 1: Fundamentos Teóricos .....	15
1.1    Botánica: .....	15
1.2    Herbario.....	15
1.3    Colección Botánica .....	15
1.4    Herbarios en el Ecuador .....	16
1.5    Herbario QCA .....	17
1.5.1    Funciones del Herbario QCA .....	17
1.5.2    Estructura del Herbario QCA .....	17
1.6    Ingeniería de Software:.....	18
1.6.1    ¿Qué es Ingeniería de Software? .....	18
1.6.2    ¿Qué es el Proceso de Software? .....	18
1.6.3    Ciclos de vida del desarrollo de software.....	19
1.7    Metodologías .....	20
1.7.1    Tipos de metodologías: .....	21
1.7.2    Metodologías tradicionales: .....	21
1.7.3    Metodologías ágiles: .....	22

1.8	Modelado UML: .....	22
1.8.1	¿Qué es un modelo? .....	22
1.8.2	¿Para qué sirve modelar?.....	23
1.8.3	¿Qué es UML? .....	23
1.8.4	Conceptos de UML: .....	23
1.8.5	Tipos de Diagramas:.....	24
1.9	Lenguajes de programación: .....	25
1.10	Herramientas a utilizar: .....	26
1.10.1	.Net Framework .....	26
1.10.2	Entity Framework.....	26
1.10.3	ADO. NET .....	26
1.10.4	Archivos edmx .....	27
1.10.5	Visual Studio 2015.....	27
1.10.6	HTML .....	27
1.10.7	JavaScript.....	27
1.10.8	Bootstrap .....	28
1.10.9	CSS.....	28
1.10.10	AngularJS .....	28
1.10.11	C# .....	29
1.10.12	Web Services.....	29
1.10.13	Rest .....	30
1.10.14	Json .....	30
1.10.15	Base de datos: .....	31
1.10.16	Sistema Gestor de Base de Datos .....	31
1.10.17	Gestor de Bases de Datos: .....	32
1.10.18	Business Intelligence .....	32
1.10.19	QlikView .....	33

1.10.20	Microsoft Azure .....	33
1.10.21	Modelo Vista Controlador .....	34
1.10.22	ASP. Net MVC .....	35
1.11	Metodología de desarrollo del Proyecto.....	35
1.11.1	Definición de Scrum.....	35
1.11.2	Transparencia.....	36
1.11.3	Inspección .....	36
1.11.4	Adaptación .....	36
1.11.5	El Equipo Scrum.....	37
1.11.6	El Product Owner .....	37
1.11.7	El equipo de desarrollo .....	38
1.11.8	Tamaño del equipo de desarrollo.....	39
1.11.9	El Scrum Master .....	39
1.11.10	Eventos de Scrum.....	40
1.11.11	El Sprint .....	41
1.11.12	Planificación del Sprint.....	42
1.11.13	Daily Scrum.....	44
1.11.14	Revisión del Sprint .....	45
1.11.15	Sprint Retrospective.....	46
1.11.16	Artefactos de Scrum.....	46
1.11.17	Pila del Producto.....	47
1.11.18	Pila del Sprint.....	48
1.11.19	Incremento.....	49
1.11.20	Transparencia de los Artefactos.....	49
1.11.21	Definición de "Terminado" .....	49
1.12	Scrum en el Desarrollo Web.....	50
1.13	Estándar IEEE SRS 830.....	51

Capítulo 2: Planificación del Proyecto .....	51
2.1 Situación Actual .....	51
2.2 Especificación de Requerimientos .....	52
2.3 Planificación del proyecto utilizando Scrum .....	53
2.4 Historias de Usuario .....	54
2.5 Recursos y Costos del Proyecto .....	57
2.6 Diagramas de Caso de Uso .....	60
2.6.1 Diagrama de Casos de Uso General .....	60
2.6.2 Diagrama F1: Gestión Accesos al Sistema .....	61
2.6.3 Diagrama F2: Gestión de Taxones .....	67
2.6.4 Diagrama F3: Gestión de Localidades .....	72
2.6.5 Diagrama F4: Gestión de Especímenes .....	77
2.6.6 Diagrama F5: Gestión de Publicaciones Científicas.....	82
2.6.7 Diagrama F6: Gestión de Préstamos .....	88
2.6.8 Diagrama F7: Gestión de Contenidos del Sitio .....	92
2.7 Diagramas de Actividades .....	98
2.8 Estándares de Codificación .....	99
2.8.1 Atributos .....	99
2.8.2 Clases .....	99
2.8.3 Controladores .....	99
2.8.4 Vistas .....	100
2.8.5 Métodos.....	100
2.8.6 Organización de archivos y carpetas para Página Web.....	100
Capítulo 3: Desarrollo del Sistema.....	101
3.1 Arquitectura .....	101
3.2 Plataforma .....	102
3.3 Diseño de interfaces .....	103

3.4	Diagramas de paquetes.....	107
3.5	Diagramas de despliegue .....	107
3.6	Sprint 1 .....	108
3.6.1	Descripción.....	108
3.6.2	Pila del producto Sprint 1 .....	108
3.6.3	Actividades del Sprint 1: .....	109
3.6.4	Pruebas de la Sprint 1 .....	110
3.7	Sprint 2.....	112
3.7.1	Descripción.....	112
3.7.2	Pila del producto Sprint 2.....	112
3.7.3	Actividades del Sprint 2: .....	113
3.7.4	Pruebas de la Sprint 2 .....	115
3.8	Sprint 3.....	117
3.8.1	Descripción.....	117
3.8.2	Pila del Producto Sprint 3 .....	118
3.8.3	Actividades de Sprint 3.....	119
3.8.4	Pruebas de Sprint 3.....	121
3.9	Sprint 4.....	123
3.9.1	Descripción.....	123
3.9.2	Pila del Producto Sprint 4 .....	123
3.9.3	Actividades del Sprint 4 .....	124
3.9.4	Pruebas de Sprint 4.....	126
3.10	Sprint 5.....	127
3.10.1	Descripción.....	127
3.10.2	Pila del Producto Sprint 5 .....	127
3.10.3	Actividades de Sprint 5.....	128
3.10.4	Pruebas de Sprint.....	129

3.11	Modelo de Base de Datos Final .....	130
3.12	Diccionario de datos .....	131
Capítulo 4: Migración de Base de Datos del HerbarioQCA .....		132
4.1	Análisis de Base de Datos en FileMaker.....	132
4.2	Diseño de la migración de información .....	134
4.3	Ejecución de la Migración de Información.....	136
4.4	Verificación de la Migración y Consultas Complejas .....	137
Capítulo 5 .....		141
1.	Conclusiones:.....	141
2.	Recomendaciones: .....	142
Anexos .....		144
Bibliografía.....		145

## Índice de Tablas

Tabla 1 Tabla comparativa entre número de ejemplares de los Herbarios del Ecuador.....	16
Tabla 2 División de Especies del Herbario.....	17
Tabla 3 Formato de Historia de Usuario.....	53
Tabla 4 Historia de Usuario Autenticación de Usuarios.....	54
Tabla 5 Historia de Usuario Gestión de Taxonomía.....	54
Tabla 6 Historia de Usuario Gestión de Contenido de las Paginas .....	55
Tabla 7 Historia de Usuario Gestión de Localidad .....	55
Tabla 8 Historia de Usuario Gestión de Espécimen .....	56
Tabla 9 Historia de Usuario Gestión de Publicaciones.....	56
Tabla 10 Historia de Usuario Gestión de Préstamos.....	57
Tabla 11 Costos detallados Servicio Microsoft Azure.....	58
Tabla 12 Software usado en el desarrollo del proyecto.....	59
Tabla 13 Detalle Actores del Sistema .....	60
Tabla 14 Pila del Producto Sprint 1 .....	108
Tabla 15 Pila de actividades Sprint 1 .....	110
Tabla 16 Pruebas Gestión de Accesos al Sistema .....	111
Tabla 17 Pruebas Gestión de Contenido del Sitio .....	111
Tabla 18 Pila del Producto para Sprint 2 .....	113
Tabla 19 Pila de actividades Sprint 2 .....	115
Tabla 20 Pruebas Gestión de Taxones .....	116
Tabla 21 Pruebas Gestión de Localidades .....	117
Tabla 22 Pila del Producto para Sprint 3 .....	118
Tabla 23 Pila de Actividades del Producto Sprint 3 .....	120
Tabla 24 Pruebas de Gestión de Publicaciones .....	121
Tabla 25 Pruebas de Gestión de Taxones.....	122
Tabla 26 Pila del Producto Sprint 4 .....	123
Tabla 27 Actividades de Sprint 4 .....	125
Tabla 28 Pruebas de Gestión de Préstamos .....	126
Tabla 29 Pruebas de Gestión de Especímenes .....	127
Tabla 30 Pila del Producto Sprint 5 .....	127
Tabla 31 Actividades de Sprint 5 .....	128
Tabla 32 Pruebas de Consulta Especímenes .....	129

## Índice de Ilustraciones

Ilustración 1 Distribución de especies del Herbario QCA .....	18
Ilustración 2 Ciclo Vital en Cascada por Vida, A. (2012). .....	20
Ilustración 3 Modelo Vista Controlador. OAF MVC Architecture. (2012) .....	34
Ilustración 4 Flujo de Proceso de SCRUM. Guía de Ingeniería del Software (2009) .....	50
Ilustración 5 Arquitectura del Proyecto.....	101
Ilustración 6 Tecnología Utilizada (Omar N., 2017).....	102
Ilustración 7 Interface Inicio de Sesión.....	103
Ilustración 8 Interface Formulario.....	104
Ilustración 9 Interface Modal.....	105
Ilustración 10 Interface Búsqueda y Resultado .....	106
Ilustración 11 Base de Datos FileMaker Herbario QCA.....	132
Ilustración 12 Base de datos 'Specimens' .....	133
Ilustración 13 Base de datos 'SciNames' .....	134
Ilustración 14 Migración de Base de Datos .....	135
Ilustración 15 Interface Cubo Taxonomía 1 .....	138
Ilustración 16 Interface Cubo Taxonomía 2.....	138
Ilustración 17 Interface Búsqueda Cubo Especimen.....	139
Ilustración 18 Interface Búsqueda Cubo Espécimen 2.....	139
Ilustración 19 Interface Búsqueda Cubo Localidades.....	140
Ilustración 20 Interface Búsqueda Cubo Publicaciones.....	140

## Índice de Diagramas

Diagrama 1 Casos de Uso General del Sistema .....	60
Diagrama 2 Caso de Uso Gestión Accesos al Sistema.....	61
Diagrama 3 Caso de Uso Gestión de Taxones .....	67
Diagrama 4 Caso de Uso Gestión de Localidades.....	72
Diagrama 5 Caso de Uso Gestión de Especímenes .....	77
Diagrama 6 Caso de Uso Gestión de Publicaciones Científicas .....	82
Diagrama 7 Caso de Uso Gestión de Préstamos.....	88
Diagrama 8 Caso de Uso Gestión de Contenidos del Sitio .....	92
Diagrama 9 Actividades de Proceso Recolección y Almacenamiento de Especímenes.....	98
Diagrama 10 Paquetes AngularJS WebApiQCAZ.....	107
Diagrama 11 Despliegue del Sistema .....	107
Diagrama 12 Base de Datos Museo .....	130

## Prefacio

La elaboración de la disertación presentada a continuación, surge como la necesidad del Herbario QCA de mejorar el proceso de almacenamiento y tratamiento de información de su colección biológica, y poder exponer estos datos al mundo a través de plataformas digitales, además de ocupar plataformas que ofrezcan mayor seguridad y confiabilidad para evitar la pérdida de información de sus muestras.

El Herbario QCA se encuentra dirigido, desde 2012, por Katya Romoleroux y Álvaro Pérez, Susana León, Priscilla Muriel como curadores.

Según datos de la información oficial del Herbario ("Información General Herbario QCA", 2016), El Herbario QCA es uno de los más grandes del país con más de 200.000 ejemplares. Las colecciones más importantes son las de helechos, palmas, gramíneas, las de árboles y arbustos del bosque andino y la Amazonía Ecuatoriana.

En la actualidad se cuenta ya con una colección representativa de Briofitas, especialmente de musgos.

El Herbario QCA tiene como objetivos:

- Conservar especímenes botánicos como un referente histórico de la Flora del Ecuador
- Contribuir a la investigación y al desarrollo de los proyectos mediante el uso de la colección
- Aportar con información de las colecciones para la publicación de libros y artículos

Con esta disertación, esperamos aportar a que la información biológica del Herbario QCA, pueda ser manejada de manera más ágil, segura y organizada, mediante la aplicación de tecnologías Web y un Motor de Base de Datos Robusto que pueda brindar seguridad al tratamiento de los datos de la institución, de manera que la colección de muestras pueda seguir creciendo ordenadamente y sirva para futuras investigaciones y exposiciones de la biodiversidad del Ecuador al mundo.

## Antecedentes

Hoy en día, las instituciones educativas, en especial aquellas dedicadas a la investigación científica, tienen una gran cantidad de información almacenada en bases de datos, esta información es de utilidad para la realización de proyectos, artículos, investigaciones, entre otras cosas.

El procesamiento de una muestra en el Herbario QCA se realiza, en términos generales, de la siguiente manera: Recolección y Prensado del Material en el Campo, Identificación por Comparación, Ingreso de cada muestra en la Base de Datos.

El Herbario QCA de la PUCE cuenta con una vasta cantidad de información, con una colección de alrededor de 218000 especímenes entre: Algas, Hélicas, líquenes, Angiospermas, Briofitas, Antoceros, Licofitas y Helechos, Gimnospermas.

Las muestras recolectadas se encuentran en las instalaciones del laboratorio donde los investigadores pueden estudiarlas y obtener información relevante para ellos.

Sin embargo, esto causa que la información no sea accesible para todos, en especial para la gente que se encuentra fuera de Quito por la falta de medios digitales, como portales web, disponibles para su exposición.

La falta de difusión de las muestras no permite el máximo aprovechamiento de información por parte los investigadores en el mundo provocando la pérdida de valor de las investigaciones.

Algunos Herbarios del Ecuador, como es el caso del Herbario de la Universidad del Azuay, cuentan con un portal web el cual pone a disposición del Mundo la información sobre las diferentes colecciones de especímenes y especies, lo que favorece al aprovechamiento de las investigaciones que se realizan en el establecimiento.

## Justificación

El Herbario QCA ha recolectado y estudiado miles de plantas desde su existencia. La información documentada en los estudios se encuentra almacenada en las instalaciones del mismo.

Actualmente, el Herbario cuenta con un proceso de muestra el cual, en resumen, empieza por la recolección y prensado del material, identificación por comparación, ingreso de cada muestra en la base de datos e ingreso de las muestras dentro de una colección.

El proceso de recolección de información de las muestras requiere de varios recursos, como económicos y humanos que hacen posible mantener gran diversidad de especies y especímenes en el herbario. Esta gran variedad de muestras es el resultado del gran esfuerzo de los colaboradores e investigadores, sin embargo, todo el esfuerzo puesto en la recolección y tratamiento de la información se encuentra desaprovechado por la falta de difusión.

Esto se puede contrarrestar mediante la creación de un portal web para consulta e ingreso de las muestras del Herbario puesto que, actualmente, su existencia es nula y la información almacenada en la base de datos solo está a disposición del Herbario, lo cual es una desventaja para otras personas que deseen conocer más sobre las plantas estudiadas.

Además, la información se maneja con una base de datos llamada FileMaker que no tiene el rendimiento y las ventajas esperadas por lo que es necesario reemplazarla con otra que mejore el desempeño y rapidez de consultas.

Por este motivo, el desarrollo de una página web junto con una base de datos robusta que despliegue y almacene la información del herbario, sería una gran herramienta para los investigadores o cualquier persona interesada en conocer sobre las especies de plantas en el Ecuador, ya que el Herbario QCA cuenta con convenios y colaboraciones con cerca de 143 herbarios nacionales e internacionales para investigación e intercambio de especímenes y ha realizado una serie de publicaciones importantes.

Dichos beneficios podrían estar a disposición de quién las necesite de manera rápida, fácil y eficiente con la existencia de dicho portal.

El desarrollo del portal web incluirá también una interfaz para la administración del ingreso de información de manera que se pueda tener un proceso establecido y seguro para el ingreso de nuevas especies de plantas.

## Objetivo General

Desarrollar una aplicación web para la administración y despliegue de información de muestras de especies y especímenes del Herbario QCA de la PUCE.

## Objetivos Específicos

- Elaborar el documento con la fundamentación teórica para el desarrollo del sistema
- Identificar los requerimientos del sistema a través de metodologías de desarrollo ágil para determinar las funcionalidades de la aplicación y su posterior documentación
- Elaborar el documento de diseño del sistema como una guía para el desarrollo del mismo
- Crear un plan de integración y pruebas mediante un documento para verificar el funcionamiento de los módulos
- Codificar los módulos de acuerdo al diseño propuesto para su posterior integración
- Realizar la integración de los módulos de acuerdo al plan de integración para asegurar el correcto funcionamiento del sistema
- Realizar las pruebas del sistema verificando el cumplimiento de las funcionalidades para la búsqueda de posibles defectos

## Alcance

El presente trabajo termina con la entrega de:

1. Documentación técnica
2. Manual de usuario
3. Sistema web con los módulos siguientes:

### **Módulo de Administración**

- Permitirá a los miembros del herbario la administración de información en la base de datos para el ingreso, modificación y eliminación del contenido.

### **Módulo de Consulta**

- Permitirá Consultar y desplegar información sobre las muestras
- Utilizará de servicios de geo posicionamiento para mostrar la ubicación de las diferentes plantas recolectadas.

### **Módulo de Reportes**

- Permitir la creación de reportes en base a la consulta realizada por el usuario

## Capítulo 1: Fundamentos Teóricos

### 1.1 Botánica:

Es la ciencia encargada del estudio de las plantas. “Se acerca a la recta interpretación del significado y evolución de los procesos vitales, así como al conocimiento de los orígenes de las plantas y de la vida misma” (Montaner y Simón, S.A. Editores 1979). Es la ciencia que se ocupa de estudiar las plantas y se integra con la Zoología y la Antropología, La Biología o ciencia que estudia la vida según Valla (1990).

Otro autor FontQuer (1985) afirma que “la botánica es la ciencia que se ocupa de todo lo referente a las plantas”.

### 1.2 Herbario

Es una herramienta de gran importancia para el estudio de la flora, en otras palabras, es un banco de datos sobre la flora de una localidad, región o país en el cual se archivan colecciones y ejemplares los cuales son secados o preservados en medios líquidos mediante diversas técnicas para luego ser ordenados de acuerdo a un reconocido sistema taxonómico.

Un herbario representa un muestrario de la distribución geográfica, características únicas e historia de los vegetales de una determinada región, esto también permite la identificación de nuevas colecciones y otros estudios.

Es importante realizar una identificación correcta de cada ejemplar porque a menudo se utilizan para actividades científicas o tecnológicas y una incorrecta identificación puede invalidar el trabajo realizado en las investigaciones.

Cerón Martínez, C. (2015). *Bases para el estudio de la flora ecuatoriana* (p. 39). Quito: Editorial Universitaria.

### 1.3 Colección Botánica

Las muestras botánicas se almacenan en estantes ya sean metálicos o de madera. Algunos de estos estantes son manuales y otros son automáticos. Usualmente el ordenamiento se realiza alfabéticamente por familias, géneros y especies.

Estas colecciones son de suma importancia, ya que contribuyen al trabajo de investigadores e investigaciones que están trabajando en el campo de la botánica.

En una colección se encuentran, además de los especímenes, varias secciones como jardín botánico, bibliotecas, salas de conferencia y muestras.

#### 1.4 Herbarios en el Ecuador

El Ecuador es un país diverso en fauna y flora, en el cual existen 13 herbarios, los cuales tienen el siguiente número de ejemplares según Cerón Martínez (2012):

Nombre	Número de Ejemplares	Ubicación
Herbario Nacional del Ecuador QCNE	230.000	Quito
Pontificia Universidad Católica del Ecuador-QCA	210.000	Quito
Universidad Central del Ecuador-QAP	80.400	Quito
Universidad Central del Ecuador-Q	21.000	Quito
Universidad San Francisco de Quito-QUSF	16.000	Quito
Padre Luis Sodiro-QPLS	13.291	Quito
Universidad de Loja	40.000	Loja
Universidad Técnica Particular de Loja-UTPL	7.000	Loja
Universidad Estatal de Guayaquil-GUAY	13.800	Guayaquil
Universidad del Azuay-HA	7.722	Cuenca
Escuela Politécnica del Chimborazo-CHEP	15.000	Riobamba
Universidad Técnica del Norte-IMAS	800	Ibarra
Galápagos-CDS	35.000	Galápagos

Tabla 1 Tabla comparativa entre número de ejemplares de los Herbarios del Ecuador. Cerón Martínez, C. (2015)

## 1.5 Herbario QCA

El herbario QCA de la Pontificia Universidad Católica del Ecuador es uno de los más grandes del país, sus colecciones más importantes son las de helechos, palmas, gramíneas y las de árboles y arbustos del bosque andino y la Amazonía Ecuatoriana. En la actualidad se cuenta ya con una colección representativa de Briofitas, especialmente de musgos.

Se ha encontrado en constante crecimiento desde sus inicios hasta la actualidad, contando actualmente con 210.000 ejemplares.

Desde hace tres décadas ha colaborado con botánicos daneses de la Universidad de Aarhus.

### 1.5.1 Funciones del Herbario QCA

Las funciones del Herbario QCA consisten, principalmente en:

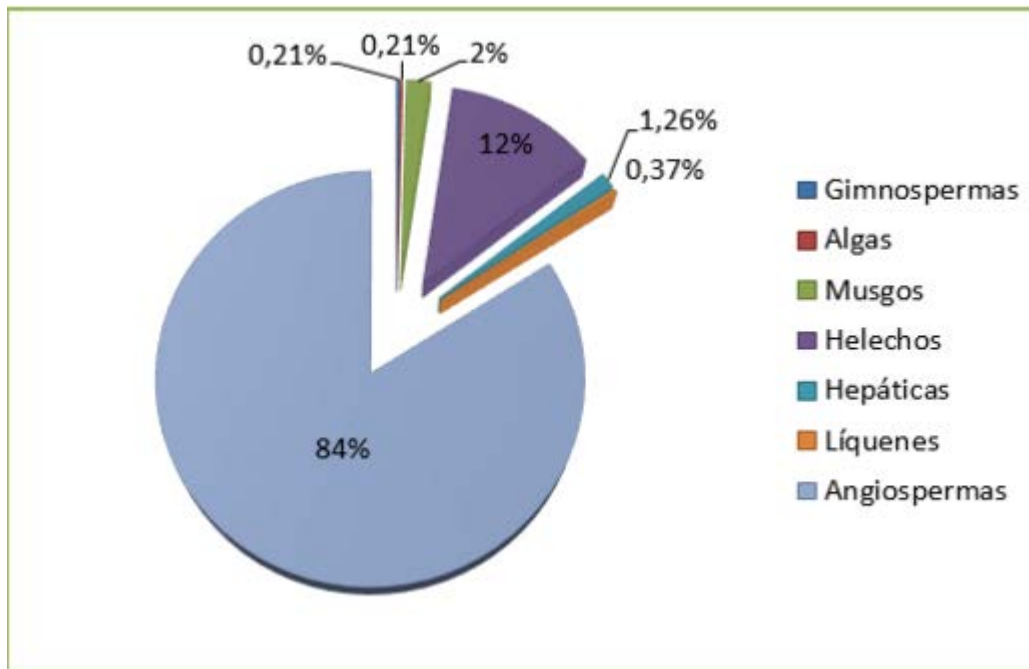
- En colaboración con otras instituciones y universidades, asistir en el desarrollo científico de estudiantes y científicos al proveer la información necesaria para proyectos enfocados en la conservación e investigación científica.
- Además, busca constituir una fuente de información para la elaboración de publicaciones.
- Mantener una colaboración frecuente con otros herbarios al realizar intercambios de especímenes y en la participación de proyectos de investigación con muchas universidades y herbarios.
- Desde 1983, el herbario ha compartido especímenes botánicos con 143 herbarios nacionales e internacionales.

### 1.5.2 Estructura del Herbario QCA

El Herbario posee 213.309 especímenes, estas se encuentran divididas entre:

Algas	Briofitas
Héticas	Antoceros
Líquenes	Licofitas y Helechos
Angiospermas	Gimnospermas

*Tabla 2 División de Especies del Herbario*



*Ilustración 1 Distribución de especies del Herbario QCA*

## 1.6 Ingeniería de Software:

### 1.6.1 ¿Qué es Ingeniería de Software?

La ingeniería de Software representa la herramienta más útil para los desarrolladores de sistemas de información, es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software a través de etapas que se realizan secuencialmente para elaborar una aplicación, desde la especificación del sistema hasta el mantenimiento de la misma. Las etapas están orientadas a obtener determinado producto como: especificaciones, diagramas, formatos, codificación, pruebas y documentación.

### 1.6.2 ¿Qué es el Proceso de Software?

De acuerdo con Sommerville (2005, p. 6), el proceso de software “es un conjunto de actividades y resultados asociados que producen un producto de software”. Existen cuatro actividades más comunes dentro de todos los procesos de software. Estas actividades son:

**Especificación del Software:** Los clientes e ingenieros definen el software a producir y las restricciones sobre su operación.

**Desarrollo del Software:** La aplicación es diseñada y desarrollada.

Validación del Software: Se realiza la validación del software para asegurar que es lo que el cliente requiere.

Evolución del Software: El software es modificado para adaptarlo a los cambios requeridos por el cliente y el mercado.

### 1.6.3 Ciclos de vida del desarrollo de software

Para construir un sistema de información es necesario tener un rumbo adecuado en el trabajo que garantice el éxito del proyecto. Por esta razón se han creado modelos llamados denominados “ciclo de vida” los cuales se han actualizado para reflejar las etapas de desarrollo involucradas y la documentación requerida, de manera que cada etapa se valide antes de continuar con la siguiente etapa en un orden cronológico.

Los modelos de desarrollo de software se han utilizado por mucho tiempo, desde la época en que el usuario de software era el programador y el problema a resolver estaba bien definido.

El modelo más antiguo y más practicado es “codifica y corrige”, el cual consiste de un lazo: Mientras

No satisface

Repite

Codifica un poco

Prueba

Corrige

Fin Mientras

Otro modelo de ciclo vital es el Cascada, el cual tiene como propósito evaluar la posibilidad de efectuar un proyecto y su alcance a través del análisis el problema en forma global enunciando y evaluando alternativas de solución, costos y beneficios de la aplicación. Además, permite calendarizar, cotizar costos y fechas probables de entregas de productos.

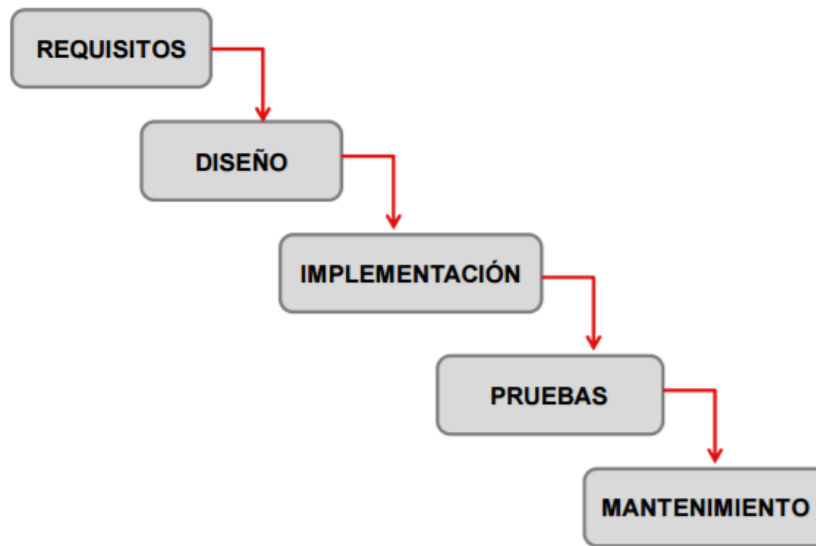


Ilustración 2 Ciclo Vital en Cascada por Vida, A. (2012).

*Análisis Comparativo de los Diferentes Ciclos de Vida. Calidaddessoftwareoyolaalex.blogspot.com. Obtenido 24/01/2017, <http://calidaddessoftwareoyolaalex.blogspot.com/2012/10/analisis-comparativo-de-los-diferentes.html>*

## 1.7 Metodologías

Las metodologías de software son una parte importante al momento de la creación de un software, por que brindan un orden y una forma de trabajar que inciden en el proceso de desarrollo.

Existen propuestas tradicionales, que se centran más en el control del proceso y actividades, además de los productos, herramientas y notaciones que van a ser usadas.

En ciertos proyectos, este tipo de metodologías han demostrado ser necesarias cuando se tienen requerimientos bien definidos, aunque en otros resulta más complicado de aplicar ya que algunos equipos necesitan métodos que presenten más enfoque en el producto de software y en el factor humano.

En ese caso los equipos deberían optar por el uso de las llamadas metodologías ágiles, que brindan mayor importancia a la interacción con el cliente, factor humano y al desarrollo incremental del producto con ciclos cortos.

Además de ser eficientes cuando los requisitos del software van a cambiar en adelante y también cuando se necesita realizar en desarrollo en tiempos reducidos.

Actualmente los equipos de desarrollo deben plantearse si usar metodologías tradicionales que hacen énfasis en la planificación o metodologías ágiles que se fijan más en la adaptabilidad del proceso.

De forma general, una metodología puede ser planteada como un conjunto de prácticas para realizar una determinada actividad de forma que esta pueda ser lograda de forma satisfactoria cumpliendo los objetivos planteados.

Aplicando la definición para desarrollo de software, según el Laboratorio Nacional de Calidad del Software de INTECO (2009), “una metodología para el desarrollo de software es marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información”.

Tomando en cuenta las definiciones anteriores, se puede decir que una metodología se basa en la unión de modelos de procesos, los cuales definen los roles, actividades, productos entregables que ofrecen al proyecto de desarrollo del software altas posibilidades de éxito. Comprenden procesos a seguir desde la concepción del proyecto hasta la finalización del mismo, es decir, cuando se ha cumplido con el propósito o propósitos por los cuales fue concebido.

#### 1.7.1 Tipos de metodologías:

Como se comentaba anteriormente, la metodología a utilizar depende mucho del tipo de proyecto de software que se va a realizar, la correcta elección de una metodología es trascendental para el éxito del proyecto.

Las metodologías de desarrollo del software pueden ser clasificadas en dos tipos. Las metodologías tradicionales y las metodologías ágiles.

#### 1.7.2 Metodologías tradicionales:

Son aquellas que centran su atención en la planificación total del trabajo a realizar, imponen una disciplina de trabajo de manera que todo el proceso esté bien detallado antes de empezar con el ciclo de desarrollo del producto de software.

Suelen ser llamadas metodologías pesadas, debido a que requieren de documentación exhaustiva del proyecto, además de que no ofrecen muchas y no es recomendable para proyectos susceptibles a futuros cambios de requerimientos.

Algunos ejemplos de este tipo de metodología son:

- RUP (Rational Unified Process)
- MSF (Microsoft Solution Framework)
- TSP (Team Software Process)
- Iconix (Pesada – Ligera)

### 1.7.3 Metodologías ágiles:

Se denominan ágiles debido a que proponen respuestas rápidas a los cambios que se pudieran dar en los requerimientos del proyecto y la fácil adaptación de los procesos de desarrollo.

Nacen como alternativa a los métodos tradicionales. Consisten en la presentación de pequeños entregables y comunicación constante con los clientes mediante el trabajo cooperativo entre desarrolladores y el cliente, la planificación es adaptativa y se da más importancia a la adaptación a un cambio que al seguimiento del plan.

Una ventaja considerable a comparación con los métodos tradicionales, es la carga de trabajo y que resulta fácil de aplicar para personas que no tienen una experticia en el seguimiento de procesos.

Algunos ejemplos de este tipo de metodología son:

- XP (Extreme Programming)
- SCRUM
- Crystal Clear
- ASD (Adaptive Software Development)
- RAD (Rapid Application Development)

## 1.8 Modelado UML:

### 1.8.1 ¿Qué es un modelo?

Antes de todo, es preciso dar una definición de 'Modelo', un modelo es una simplificación de un objeto o entidad real, que puede usarse como guía para la consecución de productos o actividades con fines similares.

## 1.8.2 ¿Para qué sirve modelar?

Citando textualmente la definición de J. Rumbaugh, I. Jacobson, G. Booch. (1999):

"Para captar y enumerar exhaustivamente los requisitos y el dominio de conocimientos, de forma que todos los implicados puedan entenderlos y estar de acuerdo con ellos".

Lo que nos permite concluir que, en resumen, un modelo es útil para obtener los requerimientos de un dominio, su división y la forma en que será usado, de manera que pueda ser entendido por los implicados en el proyecto.

## 1.8.3 ¿Qué es UML?

UML viene de las siglas '*Unified Modeling Language*' o en español, Lenguaje Unificado de Modelado.

Fue creado en 1994 por Grady Booch, James Rumbaugh e Ivan Jacobson, UML, según la definición de J. Rumbaugh, I. Jacobson, G. Booch. (1999), "es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Captura decisiones y conocimientos sobre los sistemas que se deben construir".

UML tiene como objetivo mejorar la experiencia de modelado principalmente en procesos de desarrollo iterativo y, además, está pensado para ser usado junto con herramientas de modelado visual que presenten generadores de código e informes para dar apoyo a los procesos de desarrollo orientados a objetos.

## 1.8.4 Conceptos de UML:

### 1.8.4.1 Vista:

Una vista es un subconjunto que muestra representaciones de los aspectos de un sistema.

En el nivel superior, se divide en clasificación estructural, comportamiento dinámico y gestión de modelo.

### 1.8.4.2 Clasificación estructural:

Se encuentran los elementos del sistema y sus relaciones, incluyen: Clases, casos de uso, componentes, nodos y elementos que serán base para el comportamiento dinámico.

Están clasificadas en: vista estática, vista de casos de uso y vista de implementación.

Los conceptos básicos estructurales de orientación objetos son:

- Clase
- Atributo
- Operación
- Relaciones (Generalización, Asociación y Dependencia)
- Interfaces

#### 1.8.4.3 Comportamiento dinámico:

Representa el comportamiento del sistema en el tiempo, esto incluye los cambios, vista de máquina de estados, vista actividad y vista de interacción.

#### 1.8.4.4 Gestión del modelo:

Representa la organización de los modelos de manera jerárquica. El paquete se usa como unidad genérica de organización. La gestión del modelo se organiza con otras vistas para organizar el trabajo de desarrollo y la configuración.

### 1.8.5 Tipos de Diagramas:

#### 1.8.5.1 Vista estática:

Diagrama de clases: Es una representación del dominio de un problema, con entidades, las cuales tienen relaciones entre si y el sentido del diagrama lo da la persona que lo construye.

#### 1.8.5.2 Vista de casos de uso:

Diagrama de Casos de uso: Este diagrama representan a los actores y a los procesos principales que intervienen en un proyecto de software.

#### 1.8.5.3 Vista de interacción:

Diagrama de secuencia: Se usan como representación de objetos de SW junto con su intercambio de mensajes dispuestos en una secuencia temporal, cada objeto tiene su "línea de vida", que se muestra como una línea vertical.

Diagramas de colaboración: Es similar al diagrama de secuencia, muestran la forma en que se transmiten mensajes y, como su nombre lo dice, colaboran entre sí para la implementación de una operación.

#### 1.8.5.4 Vista de la máquina de estados:

Diagramas de estados: Representar cómo evoluciona la vida de un objeto de clase, sus estados están conectados por transiciones. Cuando ocurre un evento, se desencadena una transición que transporta al objeto a otro estado.

#### 1.8.5.5 Vista de actividades:

Diagramas de actividades: Es similar a un diagrama de estados, muestra las actividades implicadas en un flujo de trabajo hasta lograr un objetivo.

#### 1.8.5.6 Vista física:

Diagrama de componentes: Muestra las dependencias y los componentes del sistema a partir de los cuales se va a construir el producto de software.

Diagrama de despliegue: Muestra los nodos del sistema representados como un cubo, con los componentes que contienen y sus enlaces a otros nodos.

#### 1.8.5.7 Vista de gestión del modelo:

Diagrama de paquetes: Los paquetes son útiles para la recopilación de clases con características similares, un diagrama de paquetes, muestra la organización o descomposición de un proyecto de software en alto nivel, mostrando sus subsistemas y dependencias.

#### 1.8.5.8 Mockup

Los mockups son modelos o prototipos que proveen una vista previa del producto para obtener la opinión del usuario. En el desarrollo de software, se utilizan para el diseño de interfaces de usuario que se pueden realizar en un papel o mediante el uso de una herramienta computarizada.

De esta manera se puede mostrar al usuario bocetos de cómo se vería el software sin tener que programarlo. Los mockups para diseño de interfaces de usuario pueden variar desde simples dibujos en papel, imágenes realistas o interfaces de usuario parcialmente funcionales.

### 1.9 Lenguajes de programación:

"Un lenguaje de programación es un medio de comunicación entre el usuario y el ordenador formado por un conjunto de símbolos elementales (alfabeto), palabras (vocabulario) y reglas

(semántica y sintaxis). Con un lenguaje se construyen cada una de las órdenes o instrucciones para el ordenador de la secuencia de instrucciones que constituyen un programa." (A. García-Beltrán et al. ,2008, p.10).

## 1.10 Herramientas a utilizar:

### 1.10.1 .Net Framework

Es un entorno de ejecución administrado que proporciona diversos servicios a las aplicaciones en ejecución. Consta de dos componentes principales: su motor de ejecución Common Language Runtime (CLR) que controla las aplicaciones en ejecución, y la biblioteca de clases de .NET Framework, que proporciona una biblioteca de código probado y reutilizable al que pueden llamar los desarrolladores desde sus propias aplicaciones.

### 1.10.2 Entity Framework

Entity Framework sirve para hacer un mapeado relacional de objetos (ORM) que reduce la desigualdad de impedancia entre el mundo orientado a objetos de los desarrolladores de .NET Framework y el mundo de las bases de datos relacionales. Permite a los desarrolladores interactuar principalmente con el modelo conceptual de una aplicación mediante técnicas orientadas a objetos conocidas. En Entity Framework, puede trabajar con datos en forma de objetos y propiedades específicos del dominio, como clientes y direcciones de clientes, sin tener que preocuparse de las tablas de las bases de datos subyacentes y las columnas en las que se almacenan estos datos. Los desarrolladores pueden emitir operaciones de acceso a datos en el modelo conceptual y Entity Framework traduce las operaciones a acciones de bases de datos relacionales.

### 1.10.3 ADO. NET

ADO.NET es un conjunto de clases que exponen servicios de acceso a datos para programadores de .NET Framework.

Ofrece abundancia de componentes para la creación de aplicaciones distribuidas.

Constituye una parte integral de .NET Framework y proporciona acceso a datos relacionales, XML y de aplicaciones.

#### 1.10.4 Archivos edmx

Según Microsoft MSDN, un archivo .edmx, “es un XML que define un modelo conceptual, un modelo de almacenamiento y la asignación entre estos modelos. Los archivos .edmx contienen información que ADO.NET Entity Designer utiliza para presentar el modelo gráficamente.”

Los modelos .edmx se generan al crear una conexión entre una base de datos con Visual Studio.

#### 1.10.5 Visual Studio 2015

Es un entorno de desarrollo integrado creado por Microsoft exclusivo para sistemas operativos Windows. Visual Studio 2015 permite a los programadores trabajar con lenguajes tales como C++, C#, Visual Basic.NET, F#, Java, Python, Ruby y PHP.

Provee a los desarrolladores las herramientas necesarias para creación de aplicaciones, servicios web y sitios en cualquier entorno que soporte ASP.NET.

Además, permite desarrollar aplicaciones para plataformas móviles con la ayuda de Xamarin.

Actualmente esta versión trabaja con .NET Framework 4.6.

#### 1.10.6 HTML

Su nombre es HyperText Markup Language o Lenguaje de marcas de Hipertexto, es un lenguaje de marcado el cual es utilizado junto con CSS y JavaScript en el desarrollo de la mayor parte de páginas web, interfaces de usuario para aplicaciones web y varias aplicaciones móviles.

HTML se encarga de describir la estructura de las páginas web a través de tags para que los navegadores interpreten el contenido y muestren el mismo como páginas web.

Sin embargo, HTML no ofrece una buena imagen a la página web, por lo que es necesario apoyarse de las hojas de estilo CSS (Cascade Style Sheet) para mostrar el contenido con estilos.

#### 1.10.7 JavaScript

Es un lenguaje de programación interpretado, utilizado principalmente en el lado del cliente, permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

“JavaScript tiene una sintaxis similar a C y toma algunos nombres de Java, aunque tienen propósitos diferentes. Actualmente, los navegadores actuales poseen un intérprete de

JavaScript. El uso más común del lenguaje es a través de funciones embebidas o incluidas en páginas HTML, interactuando con el DOM (Document Object Model).” *JavaScript*. (2016).

Las funciones escritas en JavaScript permiten cargar nuevo contenido en la página o enviar datos al servidor con AJAX, validaciones en los formularios, realizar animaciones de elementos de la página entre otras aplicaciones multimedia como reproducción de audio y video.

JavaScript también ha tenido implementaciones de script del lado del servidor como Node.js, que se utiliza actualmente.

### 1.10.8 Bootstrap

Bootstrap es un framework o conjunto de herramientas de código abierto creado por Twitter que proveen una estructura para el diseño de sitios y aplicaciones web. Está enfocado exclusivamente al desarrollo del front-end.

“Las plantillas contienen tipografía, formularios, botones, cuadros, menús de navegación y otros elementos que se pueden utilizar junto a HTML y CSS además de librerías construidas en JavaScript.” *Bootstrap (front-end framework)*. (2016).

Los elementos de bootstrap son compatibles con los navegadores Google Chrome, Firefox, Microsoft Edge, Opera y Safari, además provee diseño web responsivo por lo que es ideal para crear aplicaciones web que reajustan su layout de manera dinámica dependiendo el tamaño de la pantalla, permitiendo visualizar los sitios o aplicaciones web en escritorio, Tablet o Smartphone.

Desde la versión 3.0, adopto la filosofía Mobile-first que enfatiza el diseño web responsivo.

### 1.10.9 CSS

CSS o Cascade Style Sheets, es un lenguaje de hoja de estilos utilizado para describir la presentación de un documento escrito con un lenguaje de marcado, como HTML. Las hojas de estilo CSS fueron creadas principalmente para separar el documento de contenido con el de su presentación.

### 1.10.10 AngularJS

AngularJS es un framework de JavaScript sostenido por Google y una comunidad para crear aplicaciones web con una sola página. Apunta a la simplificación del desarrollo y pruebas de aplicaciones web al proveer un framework en las arquitecturas MVC y MVVM.

De acuerdo con la documentación oficial de AngularJS: “El framework permite ampliar el vocabulario de HTML, permitiendo declarar vistas dinámicas. El entorno resultante es extraordinariamente expresivo, legible y rápido de desarrollar. AngularJS es totalmente extensible y funciona bien con otras librerías al adaptarse a su flujo de trabajo único y necesidades.” *AngularJS — Superheroic JavaScript MVW Framework*. (2016).

#### 1.10.11 C#

C# es un lenguaje de programación orientado a objetos desarrollado por Microsoft dentro de su plataforma .NET.

Su sintaxis está basada en C/C++, al igual que Java, pero presenta mejoras de otros lenguajes.

Según el manual de C# de Andrade, E. (2017). “C# cumple el estándar ECMA-334, el cual tiene las siguientes metas:

- Lenguaje de programación orientado a objetos simple, moderno y de propósito general.
- Inclusión de principios de ingeniería de software tales como revisión estricta de los tipos de datos, revisión de límites de vectores, detección de intentos de usar variables no inicializadas, y recolección de basura automática.
- Capacidad para desarrollar componentes de software que se puedan usar en ambientes distribuidos.
- Portabilidad del código fuente.
- Fácil migración del programador al nuevo lenguaje, especialmente para programadores familiarizados con C, C++ y Java.
- Soporte para internacionalización.
- Adecuación para escribir aplicaciones de cualquier tamaño: desde las más grandes y sofisticadas como sistemas operativos hasta las más pequeñas funciones.
- Aplicaciones económicas en cuanto a memoria y procesado.

#### 1.10.12 Web Services

Un Servicio Web es una tecnología utilizada para la comunicación de un dispositivo con otro a través de WWW (World Wide Web). El uso de HTTP se utiliza para la transferencia de texto en XML o JSON.

Según Eslava Muñoz, V. (2013) Un Web Service o Servicio Web “es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web”.

Los servicios web, contribuyen en la interoperabilidad entre aplicaciones de software, sin importar cuáles son sus propiedades o bajo que plataformas se encuentran instaladas. Además, proporcionan estándares y protocolos basados en texto, lo cual facilita acceder a su contenido.

Los servicios web utilizan típicamente interfaces web orientadas a objetos para conectarse con un servidor de base de datos a través de una aplicación web o móvil.

La creación los servicios web han permitido que varias organizaciones o empresas sin importar su ubicación geográfica combinen sus servicios fácilmente de manera integrada.

#### 1.10.13 Rest

La Transferencia de Estado Representacional o RESTful Web Servicio es una manera proveer la interoperabilidad entre sistemas computacionales a través de internet. El termino fue descrito en una tesis doctoral de Roy Fielding, uno de los principales autores de la especificación del protocolo HTTP, que en un futuro fue utilizado por la comunidad.

Rest es popular al desarrollar APIs al ser ligero en la comunicación entre el productor y el consumidor, por lo cual es utilizado con frecuencia en aplicaciones móviles, redes sociales, procesos automatizados del negocio, entre otros.

El estilo de REST Enfatiza que las interacciones entre clientes y servicios son mejoradas al tener un número limitado de verbos, GET, POST, PUT, DELETE, utilizados para las operaciones de selección, inserción, actualización y eliminación.

#### 1.10.14 Json

Según su documentación oficial, *JSON*. (2016):

*“JSON o JavaScript Object Notation es un formato ligero de intercambio de datos, basado en un subconjunto del Lenguaje de Programación JavaScript utilizando texto que es completamente independiente del lenguaje, pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java,*

*JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.*

*JSON está constituido por dos estructuras:*

- *Una colección de pares de nombre/valor. En varios lenguajes esto es conocido como un objeto, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.*
- *Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.*

*Estas son estructuras universales; virtualmente todos los lenguajes de programación las soportan de una forma u otra. Es razonable que un formato de intercambio de datos que es independiente del lenguaje de programación se base en estas estructuras”.*

Ejemplo:

```
{"empleados":[  
  {"nombre":"Daniel", "apellido":"Angulo"},  
  {"nombre":"Ana", "apellido":"Salas"},  
  {"nombre":"Diego", "apellido":"Freire"}  
]}
```

#### 1.10.15 Base de datos:

La colección de datos, también denominada base de datos, es un banco de información muy bien organizada en esquemas y tablas, que puede ser importante para una empresa. Los datos están organizados para modelar aspectos de la realidad de forma que soporten procesos que requieran información. Por lo general, estos datos se encuentran interrelacionados.

#### 1.10.16 Sistema Gestor de Base de Datos

Los sistemas gestores de bases de datos (SGBD) son aplicaciones computacionales que permiten la interacción del usuario la base de datos para capturar y analizar datos. Los SGBD están diseñados para definir, crear, realizar búsquedas y operaciones de administración de los esquemas de datos.

Otro aspecto importante de estos sistemas es que deben garantizar la fiabilidad de la información almacenada en momentos críticos como una caída del sistema o en intentos de acceso no autorizados.

## 1.10.17 Gestor de Bases de Datos:

### 1.10.17.1 Microsoft SQL Server

De acuerdo con la documentación oficial de Microsoft:

*“Microsoft SQL Server es un sistema de administración y análisis de bases de datos relacionales para soluciones de comercio electrónico, línea de negocio y almacenamiento de datos “. Biblioteca de Microsoft SQL Server. (2016).*

Este SGBD utiliza como lenguaje Transact-SQL, una implementación de SQL para manipular y recuperar datos, crear tablas y definir las relaciones entre ellas.

Según la documentación oficial de la Biblioteca de Microsoft SQL Server. (2016).

*“Microsoft SQL Server 2014 se basa en funciones de su versión anterior para mantener el rendimiento, disponibilidad y facilidad de uso para aplicaciones. Además, ofrece nuevas capacidades en memoria de la base de datos principal para el procesamiento de transacciones en línea (OLTP), almacenamiento de datos y Business Intelligence”.*

Además, Microsoft señala que:

*“SQL Server 2014 también proporciona soluciones de copia de seguridad y de recuperación ante desastres, así como de arquitectura híbrida con Windows Azure, lo que permite a los clientes utilizar sus actuales conocimientos con características locales que aprovechan los centros de datos globales de Microsoft. Además, SQL Server 2014 aprovecha las nuevas capacidades de Windows Server 2012 y Windows Server 2012 R2 para ofrecer una escalabilidad sin parangón a las aplicaciones de base de datos en un entorno físico o virtual.”* Biblioteca de Microsoft SQL Server. (2016).

### 1.10.18 Business Intelligence

En los últimos tiempos, ha existido un aumento en la creación, recolección y uso de datos, que, en muchos casos, se encuentra desaprovechada principalmente por la falta de herramientas de análisis de gran cantidad de información no estructurada.

Las empresas y negocios, han mostrado un gran interés para sacar provecho de esta información, por lo cual, se han desarrollado herramientas que permiten extraer los datos necesarios para crear estrategias y tomar las mejores decisiones pensando en el futuro.

Ejemplos de estas estrategias son: Identificar cual es el mejor cliente de una compañía, predecir el dinero que se puede invertir en los próximos meses de acuerdo a datos históricos, predecir el consumo de energía basándose en el promedio de demanda pasada, entre otros.

En conclusión, La Inteligencia de Negocio es una herramienta para las empresas que permite facilitar la toma de decisiones y generar predicciones en base a datos extraídos de diferentes fuentes.

#### 1.10.19 QlikView

QlikView, es un programa de Business Intelligence desarrollado por la empresa Qlik, que permite construir aplicaciones analíticas como soluciones flexibles listas para negocios.

Se pueden crear descubrimientos de datos orientados al negocio, como guías modificables y estrechamente gobernables.

Permite crear guías analíticas controladas que pueden ayudar a los empleados de una empresa en la experiencia de descubrimiento y toma de decisiones significativas para el negocio.

La herramienta se puede modificar a medida de las necesidades de la empresa, puede combinarse con otras aplicaciones empresariales y software de gestión de sistemas a través de APIs.

Facilita la creación de informes a través de plantillas que se pueden integrar en documentos de Microsoft Office.

#### 1.10.20 Microsoft Azure

Se trata de un servicio de computación en la nube creado por Microsoft alojado en su red global de Data Centers.

Provee software como servicio (SaaS), plataforma como servicio (PaaS) e infraestructura como servicio (IaaS), además tiene soporte para diversos lenguajes de programación, herramientas y frameworks, incluyendo, como no puede ser de otra manera, las tecnologías propias de Microsoft.

Microsoft Azure integra los servicios de nube para desarrollar, implementar, probar y Administrar aplicaciones, aprovechando el poder, eficiencia y disponibilidad que ofrece la computación en la nube.

Citando a Microsoft: “Microsoft Azure es una creciente colección de servicios en la nube integrados que los desarrolladores y los profesionales de TI utilizan para crear, implementar y administrar aplicaciones a través de nuestra red global de centros de datos. Con Azure, se obtiene la libertad de crear e implementar donde quiera, utilizando las herramientas, las aplicaciones y los marcos que prefiera.” (“¿Qué es Azure? El mejor servicio en la nube de Microsoft | Microsoft Azure”, 2017).

Además, puede administrar todos sus servicios de forma remota, ya sea desde una ubicación determinada, el portal de Azure, o mediante programación utilizando varias API's y plantillas.

### 1.10.21 Modelo Vista Controlador

El Modelo Vista Controlador es un patrón de arquitectura que divide una aplicación de software en tres partes interconectadas, separando los datos y la lógica de negocio de la interfaz de usuario y la gestión de eventos. Esta arquitectura se ha vuelto popular no solo en el desarrollo de aplicaciones para escritorio, también para diseño de aplicaciones web y móviles.

Esta arquitectura se encuentra organizada de la siguiente manera:

**Modelo:** Es la representación de la información, se encuentra encargado de manejar los datos, la lógica y reglas de la aplicación.

**Controlador:** Responde a eventos, envía comandos al modelo para actualizar su estado. También puede enviar comandos a la vista para cambiar la manera de representar los datos del modelo.

**La Vista:** Presenta el modelo en un formato de interacción, normalmente la interface de usuario, requiriendo el mencionado modelo con la información para representar su salida.

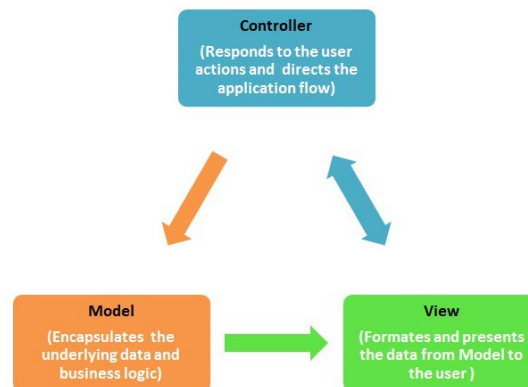


Ilustración 3 Modelo Vista Controlador. OAF MVC Architecture. (2012)

### 1.10.22 ASP. Net MVC

ASP.NET MVC es un framework ligero y altamente comprobable que se integra con las características existentes de ASP.NET.

Su diseño está basado en componentes de la aplicación, divididos en modelo, vista controlador, lo cual permite al desarrollador manejar la complejidad de proyectos de gran escala y trabajar en componentes individuales.

La estructura MVC mejora la programación orientada a las pruebas de la aplicación, ya que sus componentes pueden ser diseñados como interfaces y pueden ser probados utilizando objetos simulados.

Soporta todas las funcionalidades de ASP.NET como autorización y autenticación, MasterPages, Data Binding, Control de Usuarios, Membresías y el Routing de ASP.NET.

## 1.11 Metodología de desarrollo del Proyecto

### 1.11.1 Definición de Scrum

Scrum es un marco de trabajo para el desarrollo y mantenimiento sostenible de productos complejos.

Es un marco de trabajo con el cual las personas pueden hacer frente a problemas complejos de adaptación de manera productiva mientras se entrega productos de gran valor.

Scrum se caracteriza por ser:

- Ligero
- Sencillo de entender
- Difícil de dominar

Scrum es un marco de proceso que se ha utilizado para gestionar el desarrollo de productos complejos desde principios de 1990. Scrum es un proceso o una técnica para la construcción de productos; más bien, es un marco dentro del cual se puede emplear varios procesos y técnicas.

Scrum se compone de Equipos y sus funciones, eventos, objetos y reglas. Cada componente en el marco sirve para un propósito específico.

Las reglas de Scrum unen los eventos, las funciones y los artefactos, que rige las relaciones y la interacción entre ellos.

Scrum se basa en la teoría de control proceso empírico. El empirismo afirma que el conocimiento proviene de la experiencia y la toma de decisiones en base a lo que se conoce. la transparencia, la inspección y la adaptación son las 3 bases que la sustentan.

Scrum emplea un enfoque iterativo e incremental para optimizar el control de riesgos.

### 1.11.2 Transparencia

La transparencia significa que los proceso y sus aspectos más notables, deben ser visibles a los responsables de los resultados. Estos aspectos deben tener un estándar común, esto para que los observadores tengan un entendimiento similar de lo que está sucediendo.

Un lenguaje común, se refiere al proceso que debe ser compartido por todos los participantes. Aquellos que realizan el trabajo y aquellos que aceptan el producto del trabajo deben compartir un conocimiento común.

### 1.11.3 Inspección

En Scrum, los usuarios deben inspeccionar con frecuencia los artefactos y como avanza hacia una meta para poder detectar posibles variaciones. Estas inspecciones no deben ser tan frecuentes si la inspección se interpone en el progreso del producto. Las inspecciones son más beneficiosas cuando se realiza con celeridad por interventores capacitados.

### 1.11.4 Adaptación

Si un inspector que está realizando la inspección determina que una o varias características de un proceso se desvían de los límites aceptables, y que esto afectara al resultado del producto final, dicho material que se procesa debe ser ajustado. Este ajuste debe realizarse tan pronto como sea posible para minimizar los posibles desvíos que puedan ser causados por ese inconveniente.

Scrum establece 4 eventos para proceder con la inspección y adaptación.

- Planificación del Sprint
- Daily Scrum
- Revisión del Sprint
- Sprint Retrospective

### 1.11.5 El Equipo Scrum

El equipo de Scrum está conformado por un Product Owner, el equipo de desarrollo, y un Scrum Master.

Los Equipos de Scrum se caracterizan por ser organizados y multifuncionales. Son organizados porque pueden elegir la mejor manera para llevar a cabo su trabajo, en lugar de ser dirigido por personas externas, y multifuncionales porque tienen todas las competencias necesarias para realizar el trabajo sin depender de otros que no forman parte del equipo.

El modelo de equipo de Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad.

Los equipos de Scrum entregan productos de manera incremental e iterativa, maximizando las oportunidades para recibir retroalimentación. Las entregas incrementales del producto garantizan tener versiones potencialmente útiles y que el trabajo este siempre disponible.

### 1.11.6 El Product Owner

El Product Owner es responsable de maximizar el valor del producto y el trabajo del equipo de desarrollo. Esto puede variar con cada organización, equipos de Scrum e individuos.

El Product Owner es la única persona responsable de la gestión de la Pila del Producto, el cual incluye:

Elementos de la Pila del Producto claramente definidos.

Ordenamiento de los Elementos de la Pila del Producto para lograr los objetivos y misiones.

Optimizar el valor del trabajo que realiza el Equipo de Desarrollo.

Asegurarse de que la reserva de pedidos de productos sea visible, transparente y clara para todos, y mostrar lo que el Equipo Scrum trabajará en la próxima.

Asegurar el equipo de desarrollo comprenda los elementos de la Pila del Producto al nivel necesario.

El Product Owner puede hacer el trabajo, o delegar al Equipo de Desarrollo para hacerlo. Sin embargo, el Product Owner sigue siendo responsable.

El Product Owner es una persona, no un comité. El Product Owner puede representar los deseos de un comité en la Pila del Producto, pero aquellos que quieran cambiar la prioridad de un elemento del mismo, debe abordar el Product Owner.

Para que el Product Owner tenga éxito, la organización debe respetar sus decisiones, las cuales son visibles en el contenido y el orden de la Pila del Producto. Nadie puede decirle al equipo de desarrollo que debe trabajar con un conjunto diferente de requisitos, y no se permite que el equipo de desarrollo realice lo que digan los demás.

#### 1.11.7 El equipo de desarrollo

El equipo de desarrollo está formado por profesionales que realizan el trabajo entregar un incremento potencialmente funcional del producto al final de cada Sprint. El incremento es creado únicamente por los miembros del equipo de desarrollo.

La estructura y capacitación de los Equipos de Desarrollo es dada por la organización, para mantener y gestionar el trabajado de manera ordenada. Como resultado se optimiza la eficiencia y la eficacia general del equipo de desarrollo.

Los equipos de desarrollo tienen las siguientes características:

Son organizados. Ninguna persona debe decir al equipo de desarrollo cómo trabajar en la conversión de la Pila del Producto en incrementos funcionales.

Los equipos de desarrollo son multifuncionales, poseedor de las habilidades necesarias para crear un incremento del producto.

Scrum no reconoce títulos para los miembros del equipo de desarrollo más que desarrollador, sin importar el trabajo que se realiza por la persona.

Independientemente del dominio que deba ser tratado como pruebas o análisis de negocios, Scrum no reconoce sub-equipos en el equipo de desarrollo, independientemente de dominios concretos.

Los miembros del equipo de desarrollo pueden tener habilidades especializadas y áreas de interés, pero la responsabilidad pertenece al equipo de desarrollo en su conjunto.

### 1.11.8 Tamaño del equipo de desarrollo

El Tamaño óptimo del equipo de desarrollo es aquel lo suficientemente pequeño como para continuar siendo ágil y lo suficientemente grande como para completar un trabajo complejo en un Sprint. Normalmente se indica que, con menos de 3 miembros en el equipo de desarrollo, estos disminuyen su interacción y la productividad es menor. Además, los pequeños equipos pueden encontrarse con limitaciones de habilidad y conocimiento durante el Sprint, haciendo que el equipo no sea capaz de entregar un avance significativo.

Tener muchos miembros en el equipo de desarrollo también puede tornarse perjudicial, contar con más de 9 miembros requiere demasiada coordinación. Este tipo de equipos es más difícil de manejar y encausar. El Product Owner y el Scrum Master no son incluidos en el número de miembros a menos que también estén participando en trabajos de la Pila del Sprint.

### 1.11.9 El Scrum Master

Es la persona responsable de asegurar que la metodología Scrum sea entendida y propagada entre los miembros del equipo. Los Scrum Masters logran hacer esto asegurándose de que el Equipo tenga siempre presente la teoría de Scrum, sus prácticas y normas.

El Scrum Master es un líder para el Equipo, ayuda a las personas que están fuera del equipo Scrum entender cuáles de sus interacciones ayudan a colaborar con el equipo y cuáles no.

El Scrum Master ayuda al Product Owner de varias formas, entre las que se encuentran:

#### **Scrum Master y Product Owner**

- Encontrar técnicas eficientes para el manejo de la Pila del Producto.
- Brindar ayuda al equipo de Scrum para entender las necesidades de los elementos de la Pila de Producto.
- Hacer que la comprensión de la planificación de los productos se de en un entorno empírico.
- Asegurar que el Product Owner sabe la manera idónea de organizar la Pila del Producto.
- Procurar que los eventos Scrum se den conforme a lo necesario.

### ***Scrum Master y el Equipo de desarrollo***

El Scrum Master ayuda al Equipo de desarrollo de varias formas, entre las que se encuentran:

Actúa como entrenador del equipo de desarrollo promoviendo la auto-organización y funcionalidad cruzada.

Ayudar a que el equipo de desarrollo tenga mayor predisposición para crear productos de alto valor.

Colaborar con la eliminación de los obstáculos que puedan afectar o frenar el progreso realizado por el Equipo de Desarrollo.

Facilitar la realización de los eventos de Scrum.

Ayudar a la adaptabilidad en los entornos de la organización en donde Scrum aún no es adoptado.

### ***Scrum Master y la Organización***

El Scrum Master ayuda a la organización de varias formas, entre las que se encuentran:

Dirigir y entrenar a la organización para que Scrum sea más fácil de ser adaptado.

Planificar la implementación de Scrum dentro de la organización.

Ayudar a los empleados y partes interesadas a comprender y poner en práctica Scrum y el conocimiento empírico para el desarrollo de productos.

Encauzar los cambios de productividad del equipo Scrum.

Trabajar en conjunto con otros Masters Scrum para mejorar la eficacia de Scrum en la organización.

#### **1.11.10 Eventos de Scrum**

Scrum cuenta con eventos predefinidos los cuales permiten mantener regularidad y reducir la necesidad de reuniones. Todos los eventos han sido creados con un límite tiempo, de tal manera que cada evento tiene una duración máxima.

Al comenzar un Sprint, su duración es fija y no se puede modificar. Los eventos restantes pueden terminar al cumplir sus objetivos, lo que garantiza una cantidad apropiada de tiempo se dedica sin permitir el desperdicio en el proceso.

Además del Sprint, los eventos en Scrum ayudan a realizar inspecciones y adaptar algo de manera formal.

#### 1.11.11 El Sprint

El Sprint es un lapso de tiempo, alrededor de un mes, en el cual se crea una parte del incremento del producto potencialmente entregable. La duración de los Sprints no varía a lo largo del desarrollo. El nuevo Sprint se inicia al terminar el anterior.

Los Sprints consisten de la Planificación del Sprint, Scrum diarios, el trabajo de desarrollo, la Revisión del Sprint, y la retrospectiva de Sprint.

Durante el Sprint:

- No se realizan cambios que pondría en peligro el objetivo del Sprint.
- Los objetivos de calidad no disminuyen.
- El alcance puede aclararse y ser renegociado entre el Product Owner y el equipo que cada vez tiene mayor conocimiento.

Cada Sprint cuenta con una definición de lo que debe ser construido, un diseño y un plan flexible que guiará su construcción, el trabajo, y el producto resultante.

La duración de los Sprint se limita a un mes. Cuando el horizonte de un Sprint es muy largo, la definición de lo que se construye puede cambiar, la complejidad y el riesgo pueden aumentar. Los Sprints permiten la previsibilidad garantizando la inspección y la adaptación de progreso hacia un objetivo del Sprint por lo menos cada mes calendario. Sprints también limitan el riesgo de costo de un mes.

#### ***Cancelación de un Sprint***

Un Sprint puede ser cancelado antes de tiempo, por medio del Product Owner, el cual es el único con la autoridad para cancelarlo, aunque puede hacerlo bajo la influencia de las partes interesadas, el equipo de desarrollo, o el Scrum Master.

Un Sprint podría ser cancelado si el objetivo del Sprint se vuelve obsoleto. Esto podría ocurrir si la empresa cambia de dirección, las condiciones del mercado o la tecnología. Sin embargo, debido a la corta duración de los Sprints, es raro que una cancelación tenga sentido.

Cuando un Sprint se cancela, se revisan los elementos terminados de la Pila del Producto. Si una parte del trabajo es potencialmente liberable, el Product Owner normalmente los acepta.

El cancelar un Sprint representa pérdida de recursos, ya que cada se debe realizar nuevamente la Planificación del Sprint para iniciar. La cancelación de un Sprint a menudo es traumática para el Equipo Scrum, pero este caso es poco frecuente.

#### 1.11.12 Planificación del Sprint

El trabajo a realizar está previsto en la Planificación del Sprint, el cual se crea con la colaboración de todo el equipo Scrum.

La planificación del Sprint se lleva a cabo en un máximo de ocho horas, si su duración es de un mes. Para los Sprints cortos, el evento es más corto. El Scrum Master asegura que el evento se lleve a cabo y que los auxiliares de entender su propósito. El Scrum Master enseña el Equipo Scrum para mantenerlo dentro de la caja de tiempo.

La Planificación del Sprint responde a las siguientes preguntas:

- ¿Qué se puede entregar en el incremento resultante del próximo Sprint?
- ¿Cómo se realizará el trabajo necesario para entregar el Incremento?

#### **¿Qué se puede entregar en el incremento resultante del próximo Sprint**

El equipo de desarrollo trabaja para pronosticar la funcionalidad que se desarrollará durante el Sprint. El Product Owner discute el objetivo que se debe lograr en el Sprint y los elementos de la Product Backlog que, si se completa en el Sprint, logran el objetivo del Sprint. Todo el Equipo Scrum colabora en la comprensión de la labor del Sprint.

Las entradas para esta reunión son la Pila del Producto, el último incremento del producto, la capacidad proyectada del equipo de desarrollo durante el Sprint, y el rendimiento pasado del equipo de desarrollo. El equipo de desarrollo decide el número de elementos que seleccionará de la Pila del Producto. Sólo el equipo de desarrollo puede evaluar lo que se puede lograr en el próximo Sprint.

El objetivo del Sprint se establece después de que el equipo decida que elementos de la Pila del Producto se entregará en el Sprint. El objetivo del Sprint se debe cumplir a través de la implementación de la Pila del Producto, guiando al equipo para la construcción del incremento.

## **¿Cómo se realizará el trabajo necesario para entregar el Incremento?**

Se denomina Pila del Sprint a los elementos seleccionados de la Pila del Producto en el Sprint y su plan de entrega.

El Sprint comienza cuando el Equipo de Desarrollo realiza el diseño del sistema y el trabajo necesario para convertir la Pila del Producto en un incremento funcional. El trabajo varíe en tamaño o esfuerzo durante el Sprint, está previsto por el Equipo de Desarrollo.

El Product Owner puede ayudar a aclarar los elementos de la Pila producto seleccionado y hacer concesiones. Si el equipo de desarrollo determina que tiene el trabajo demasiado o demasiado poco, puede renegociar los elementos de la Pila del Producto seleccionado con el Product Owner. El equipo de desarrollo también puede invitar a otras personas a asistir con el fin de proporcionar asesoramiento técnico o de dominio.

Hacia el final de la Planificación del Sprint, el equipo de desarrollo debe ser capaz de explicar al Product Owner y Scrum Master cómo se propone trabajar como un equipo de auto-organización para lograr el objetivo del Sprint y crear el esperado incremento.

### **Meta del Sprint**

La Meta del Sprint es un objetivo que puede ser alcanzado al implementar la Pila del Producto, proporcionando una guía para el Equipo de Desarrollo sobre que está construyendo. Se establece en la reunión de planificación de Sprint.

El objetivo del Sprint puede hacer que el equipo de desarrollo trabaje en conjunto, a través de iniciativas independientes.

El Equipo de Scrum toma en cuenta la Meta del Sprint mientras realiza el trabajo, con el fin de satisfacerla implementando la funcionalidad. Si el trabajo realizado resulta ser diferente al esperado por el Equipo de Desarrollo, se debe colaborar con el Product Owner para negociar el alcance de Sprint Backlog en el Sprint.

### 1.11.13 Daily Scrum

El Daily Scrum es un evento de 15 minutos para que el equipo de desarrollo sincronice las actividades y cree un plan para las 24 horas siguientes.

Para realizar esto, se inspecciona los trabajos desde el último Daily Scrum y también la previsión de los trabajos que se pueden hacer hasta antes del siguiente.

El Daily Scrum se lleva a cabo en una determinada hora y lugar cada día. Durante la reunión, se tratan temas como los siguientes:

- ¿Qué hice ayer que ayudó al equipo de desarrollo a llegar a cumplir con el objetivo del Sprint?
- ¿Qué voy a hacer hoy para ayudar al equipo de desarrollo a llegar a cumplir con el objetivo del Sprint?
- ¿Qué impedimento encuentro que me impide o impide al Equipo de Desarrollo llegar a cumplir con el objetivo del Sprint?

El equipo de desarrollo utiliza el Daily Scrum para inspeccionar el progreso hacia el objetivo del Sprint y también las actividades en la Pila de Sprint.

El Daily Scrum optimiza la probabilidad que el equipo de desarrollo pueda cumplir con el Objetivo del Sprint. Cada día, el equipo de desarrollo debe entender cómo van a trabajar juntos para lograr el objetivo del Sprint y crear incrementos para el final del Sprint.

El equipo de desarrollo a menudo se reúne inmediatamente después del Daily Scrum para hablar de asuntos determinados o para planificar el resto del trabajo de la Sprint.

El Scrum Master asegura que se dé la reunión, pero el equipo de desarrollo se encarga de realizar el Daily Scrum. El Scrum Master debe enseñar al equipo de desarrollo que El Daily Scrum este dentro de los 15 minutos establecidos.

El Scrum Master debe procurar que únicamente los miembros del Equipo de Desarrollo participen en el Daily Scrum.

Los Daily Scrums sirven para mejorar comunicación, identificar obstáculos para el desarrollo, promover la toma de decisiones rápida, y mejorar el nivel de conocimiento del equipo de desarrollo.

#### 1.11.14 Revisión del Sprint

La revisión del Sprint se lleva a cabo al final de cada Sprint para inspeccionar el progreso y adaptar la Pila del Producto si es necesario. En la Revisión del Sprint, el Equipo Scrum y las partes interesadas se reúnen y revisan lo que se hizo en el Sprint.

Teniendo esa base y también cualquier cambio realizado en la Pila del Producto durante el Sprint, Los asistentes colaboran en las cosas que se podrían hacer para optimizar el valor del producto. Esta presentación, está principalmente destinada a mejorar la realimentación y fortalecer la colaboración.

La reunión suele tener un tiempo de cuatro horas para los Sprints de un mes. Para los Sprints cortos, por lo general más corta. El Scrum Master debe asegurar que el evento se llevará a cabo y que los asistentes entenderán su propósito.

Esta reunión tiene las siguientes características:

- El Equipo de Scrum y las partes interesadas son los principales asistentes.
- El Product Owner explica qué elementos de la Pila del Producto han sido Terminados y cuáles no.
- El equipo de desarrollo analiza lo que ha ido bien durante el Sprint, qué problemas se tuvieron, y que soluciones se dieron para estos problemas.
- El equipo de desarrollo demuestra el trabajo que se tiene Terminado y responde a preguntas sobre el desarrollo.
- El Product Owner revisa la Pila del Producto en su estado actual y presenta fechas de finalización tentativas basadas en los avances hasta la fecha (opcional).
- Todo el grupo colabora en qué se va a hacer continuación, para recolectar información importante para la posterior Planificación del Sprint.
- Se Revisa como el mercado o el uso potencial del producto podrían haber cambiado y que se debe hacer a continuación.
- También suele hacerse una revisión de la línea de tiempo, presupuesto y el mercado para la próxima versión del producto.

El resultado de la Revisión del Sprint es un Pila del Producto que define los posibles elementos para el próximo Sprint, este también puede ser ajustada para estar preparado ante posibles nuevas oportunidades.

#### 1.11.15 Sprint Retrospective

El Sprint Retrospective tiene como objetivo que el Equipo Scrum pueda hacer una auto inspección y crear un plan de mejoras que se revise durante el próximo Sprint.

El Sprint Retrospective se da después de la Revisión del Sprint y antes de la próxima Planificación del Sprint.

Es una reunión de tres horas para los Sprints de un mes. Para Sprints cortos es, por lo general, más corta.

El Scrum Master procura que todo se mantenga dentro del rango de tiempo. El Scrum Master también participa como miembro del equipo en la reunión.

El de la Sprint Retrospective es:

- Analizar cómo fue el último Sprint con respecto a las personas, relaciones, procesos y herramientas.
- Identificar y ordenar los principales elementos que resultaron bien y que mejoras potenciales se pueden obtener.
- Crear un plan para la implementación de mejoras a la manera en la que el Equipo hace su trabajo.

Además, el Scrum Master anima al Equipo para que mejore, dentro del marco de proceso Scrum, sus prácticas de desarrollo de manera que el proceso se torne más eficaz y agradable para el próximo Sprint. Durante cada Sprint Retrospective, el equipo Scrum planea estrategias para lograr mejorar la calidad del producto.

El Sprint Retrospective es un evento para centrarse en la inspección y la adaptación de los elementos, Al final de este, el Equipo Scrum debería haber identificado algunos elementos para ser corregidos o mejorados en el próximo Sprint.

#### 1.11.16 Artefactos de Scrum

Los artefactos de Scrum representan el trabajo o valor que provee transparencia y oportunidades para la inspección y la adaptación. Los Artefactos definidos por Scrum están diseñados específicamente para maximizar la transparencia de la información clave de forma que todos tengan la misma comprensión.

### 1.11.17 Pila del Producto

La Pila del Producto es una lista ordenada de lo que necesitara el producto y es la única fuente de requisitos para cualquier cambio que se introduzca en el producto.

El Product Owner es responsable de la Pila de Producto, incluyendo su contenido, disponibilidad, y el pedido.

La Pila del Producto nunca es completa, pero evoluciona a medida que el producto y el medio ambiente en el que se va a utilizar sufre cambios.

En la Pila del Producto se enumeran todas las características, funciones, requisitos, mejoras y correcciones que constituyen los cambios en futuras versiones del producto con una descripción, orden, estimación y valor.

Con el uso del producto, su aumento de valor y la retroalimentación del mercado, la Pila del Producto se convierte en una lista más grande y exhaustiva. Los cambios en los requerimientos del negocio, las condiciones del mercado o la tecnología pueden causar cambios en la Pila del Producto.

A menudo, varios equipos de Scrum trabajan juntos en el mismo producto. Una Pila de Producto se puede utilizar para describir el próximo trabajo en el producto.

El refinamiento de la Pila de Producto consiste en añadir detalles, estimaciones, y el orden de los elementos de la Pila de Producto. Este es un proceso en el que el Product Owner y el Equipo de Desarrollo colaboran detallando los elementos de la Pila del Producto. Los elementos de la Pila del Producto pueden ser actualizados en cualquier momento por el Product Owner o, a discreción del Product Owner.

Los elementos de la Pila con mayor orden son por lo general más claros y detallados que los de menor orden. Las estimaciones más precisas se realizan en base a mayor claridad y detalle; mientras sea menor sea el orden, menor es el detalle.

Los elementos de la Pila del Producto que realizara el Equipo de Desarrollo en el próximo Sprint se refinan para que cualquier artículo pueda ser razonablemente "Terminado" dentro de la duración del Sprint. Los elementos de la Pila de productos que han sido terminados por el equipo de desarrollo dentro de un Sprint se consideran "Listos" para la selección de una Planificación del Sprint. Estos elementos suelen adquirir este grado de transparencia a través de las actividades de refinación descritos anteriormente.

El equipo de desarrollo es responsable de todas las estimaciones. El Product Owner puede influir en el Equipo de Desarrollo ayudándolo a comprender y seleccionar soluciones, pero las personas que van a realizar el trabajo realizan la estimación final.

### **Monitoreo del progreso hacia una meta**

En cualquier punto del tiempo, el trabajo restante para alcanzar una meta puede ser resumido. El Product Owner realiza el seguimiento del trabajo con cada Revisión del Sprint. El Product Owner compara el progreso con el resto de trabajo en las Revisiones de Sprint para asegurar el avance hacia la conclusión de lo planteado en el tiempo deseado. Esta información se hace transparente para todas las partes interesadas.

Diversas prácticas proyectivas sobre tendencias se han utilizado para pronosticar el progreso, como Burn-downs, Burn-ups, o flujos acumulados. Estos han demostrado ser útiles. Sin embargo, éstas no reemplazan la importancia del empirismo, al utilizar lo que ha sucedido para tomar decisiones a futuro.

#### **1.11.18 Pila del Sprint**

La Pila del Sprint es una lista de tareas o cambios que el Equipo realizara en un Sprint, además de un plan para entregar el incremento del producto. El equipo de desarrollo prevé las funcionales que se entregaran en el siguiente incremento.

La Pila del Sprint hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para cumplir con el objetivo del Sprint.

La Pila del Sprint puede ser modificada por el equipo durante la iteración, mientras el equipo este trabajando a través del plan, aprendiendo sobre su trabajo y de esta manera cumplir el objetivo del Sprint.

Al requerir un nuevo trabajo, el equipo de desarrollo debe añadir la tarea a la Pila del Sprint. Conforme se avance el trabajo, se actualizará su estimación. Si existen elementos innecesarios pueden ser retirados. El equipo de desarrollo es el único autorizado para cambiar su Pila del Sprint en ejecución.

## **Monitoreo del Progreso del Sprint**

El equipo de desarrollo debe dar seguimiento del trabajo restante, al menos para cada Daily Scrum proyectándose la probabilidad de alcanzar el objetivo del Sprint. Mediante el seguimiento de los trabajos pendientes en todo el Sprint, el equipo de desarrollo puede gestionar su progreso.

### **1.11.19 Incremento**

El incremento es la parte de un producto producida durante un sprint, el cual debe estar terminado y ser funcional para ser liberado al cliente.

### **1.11.20 Transparencia de los Artefactos**

Scrum está basado en la transparencia. Las decisiones para optimizar el valor y controlar el riesgo se realizan basándose en el estado de los artefactos.

Cuando se tiene una transparencia completa, estas decisiones tienen una base sólida. Caso contrario, las decisiones pueden ser desencadenantes para que el valor disminuya y riesgo puede aumentar.

El Scrum Master debe trabajar en conjunto con el Product Owner, equipo de desarrollo y partes interesadas para determinar si los artefactos son lo suficientemente transparentes como se espera.

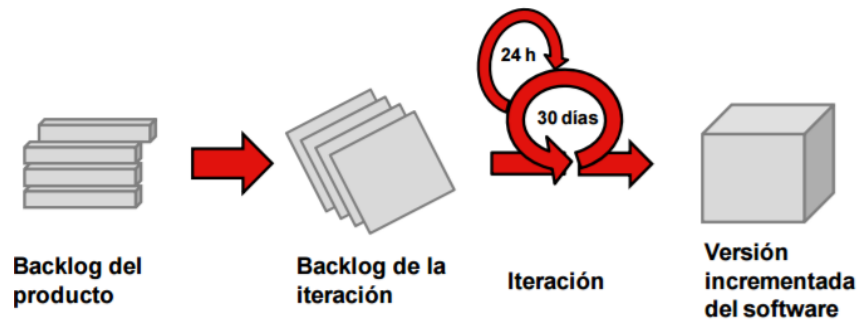
El Scrum Master puede detectar la transparencia incompleta mediante la inspección de los artefactos y detección de patrones, debe alentar a la aplicación de las prácticas más adecuadas cuando se tenga artefactos ausentes de transparencia total.

### **1.11.21 Definición de "Terminado"**

El propósito de cada Sprint es entregar incrementos de funcionalidad potencialmente liberable que se adhieren a la definición actual del Equipo Scrum de "Terminado".

Para cada producto, debe definirse cuando su estado sea "Terminado", el cual debe aplicarse a cada incremento.

La definición de "Terminado" debe ser comprendida por todo el equipo para establecer la finalización del incremento ya que el Product Owner puede optar por liberarlo inmediatamente.



*Ilustración 4 Flujo de Proceso de SCRUM. Guía de Ingeniería del Software (2009)*

## 1.12 Scrum en el Desarrollo Web

Debido a la naturaleza dinámica de los proyectos en el desarrollo web, existen algunas buenas razones para utilizar Scrum durante la ejecución de un proyecto.

- Utilizar Scrum de manera ágil permite que la compañía ahorre tiempo y dinero.
- La metodología permite que se crea camaradería entre los miembros del equipo.
- Permite desarrollar con éxito a los proyectos donde los requerimientos de negocio son difíciles de documentar debido a su tamaño.
- Se pueden realizar movimientos rápidos durante la codificación, permitiendo realizar pruebas y correcciones inmediatas.
- Es un método ligeramente controlado a través de reuniones frecuentes donde se actualiza el proyecto y se visualiza el avance del proyecto.
- Como otras metodologías ágiles, es de naturaleza iterativa, permitiendo la retroalimentación continua por parte del usuario y que el hacer frente a los cambios sea más fácil.
- Las reuniones diarias hacen posible la evaluación de productividad individual. Como resultado se obtiene mejora en la productividad de cada miembro del equipo.
- La identificación de problemas es más efectiva gracias a las reuniones diarias, permitiendo su corrección inmediata.
- Es más fácil realizar la entrega del producto en el tiempo establecido.
- Scrum puede trabajar con cualquier tecnología o lenguaje de programación, pero es particularmente útil con proyectos orientados a la web 2.0.
- El costo total en términos de progreso y administración es dando lugar a un resultado más ágil y a menor precio.

### 1.13 Estándar IEEE SRS 830

El IEEE SRS 830, es un documento estandarizado contenido de buena calidad para la especificación de requisitos de software (SRS).

En este documento se pueden detallar los requisitos funcionales y no funcionales, así como los casos de uso que describen las interacciones que el software debe proveer, permitiendo el aseguramiento riguroso de los requerimientos antes del diseño para evitar la pérdida de tiempo por el rediseño. Adicionalmente se puede establecer costos, riesgos y horarios.

## Capítulo 2: Planificación del Proyecto

### 2.1 Situación Actual

Durante varios años, el Herbario QCA de la Pontificia Universidad Católica del Ecuador se ha dedicado a la recolección y estudio de miles de plantas en nuestro país<sup>1</sup>. Las muestras han sido documentadas y almacenadas en las instalaciones del mismo.

Actualmente, el Herbario cuenta con un proceso de muestra el cual, en resumen, empieza por la recolección y prensado del material, identificación por comparación, ingreso de cada muestra en la base de datos e ingreso de las muestras dentro de una colección<sup>2</sup>.

El proceso de recolección de información de las muestras requiere de varios recursos, como económicos y humanos que hacen posible mantener gran diversidad de especies y especímenes en el herbario. Esta gran variedad de muestras es el resultado del gran esfuerzo de los colaboradores e investigadores.

En la visita realizada a las instalaciones del Herbario QCA, nos indicaron que, para mantener un registro de especímenes recolectados, se utiliza el programa FileMaker como base de datos, y una serie de formularios que permiten a los colectores ingresar nuevas especies, especímenes, definir cuál es su localización, gestionar Préstamos, agregar publicaciones y usos de cada planta.

El diseño actual de la base de datos no cuenta con una estructura bien definida, con tablas que se relacionan muchas veces de forma confusa y, en la mayoría de casos, son

---

<sup>1</sup> *Información General*. (2017). *Puce.edu.ec*. Retrieved 26 January 2017, from <http://www.puce.edu.ec/portal/content/Herbario/477?link=oln30.redirect>

<sup>2</sup> *Nuestra colección*. (2017). *Puce.edu.ec*. Retrieved 26 January 2017, from <http://www.puce.edu.ec/portal/content/NuestraColección/539?link=oln30.redirect>

innecesarias. Esto causa inconformidad por parte de los usuarios al tener un bajo rendimiento en el tratamiento de los datos corriendo el riesgo de perder los mismos por que el sistema no ofrece la solidez necesaria.

Sin embargo, todo el esfuerzo puesto en la recolección y tratamiento de la información se encuentra desaprovechado por la falta de difusión, lo cual ha generado en el herbario la necesidad de contar con una plataforma digital para exponer información relevante de las especies de su colección.

La información histórica en las áreas de herbario contiene datos inconsistentes que deben ser depurados en el actual sistema y posteriormente en el Sistema BioWeb.

Con el propósito de salvaguardar la integridad de la información, la actualización de información se basará únicamente en plantillas que contiene campos específicos definidos por el responsable de cada área.

## 2.2 Especificación de Requerimientos

Dentro de la Metodología de Desarrollo SCRUM, se establece como punto de partida la especificación de la pila del producto, la cual establece una lista priorizada de los requerimientos del proyecto.

En esta etapa se busca definir todas las funcionalidades y restricciones del sistema, permitiendo a los desarrolladores y usuario final llegar a un acuerdo sobre lo que se debe hacer.

La estimación de la pila del producto se realizó con el apoyo del Documento de Especificación de Requerimientos (SRS) IEEE 830 2008, el cual permitió definir los diferentes casos de uso del sistema, mediante el estudio de las necesidades del cliente.

El documento de requerimientos o SRS anexado, se basa en el modelo realizado por Juan Tapias (Tapias, 2016).

La metodología SCRUM no señala la utilización de este documento, sin embargo, se optó por utilizarlo para facilitar el levantamiento de requerimientos.

(Anexo 1)

## 2.3 Planificación del proyecto utilizando Scrum

Para establecer los ítems de la Pila del Producto, se utilizaron historias de usuario (User Stories), que permiten documentar las necesidades de las partes interesadas en una serie de reuniones que se realizaban cada dos semanas.

Para realizar las historias de usuario, se utilizó una plantilla basándose en el esquema utilizado en la Tesis de Chávez, A, & Tenorio, J. (2012). con los siguientes campos:

- **ID:** Identificado único de la historia de usuario.
- **Nombre:** Es una descripción corta de la historia.
- **Importancia:** Definido por el propietario del producto, es el grado de prioridad para el desarrollo de cada historia de usuario.
- **Estimación Inicial:** Es la estimación inicial del equipo sobre la cantidad de trabajo que es necesario para implementar la historia, comparada con otras historias.
- **Como Probarlo:** Una descripción a alto nivel de cómo se demostrará esta historia en la Demo al final del Sprint.
- **Notas:** Cualquier otra información, clarificación, referencia a otras fuentes de información, etc.

Historia de Usuario			
Número		Usuario	
Nombre de la Historia			
Prioridad en Negocio			
Descripción			
Como Probarlo			

*Tabla 3 Formato de Historia de Usuario*

## 2.4 Historias de Usuario

Historia de Usuario			
<b>Número</b>	1	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Autenticación de Usuarios.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Se requiere que el sistema permita ingresar solo a usuarios registrados en el mismo mediante un nombre y una contraseña, y pueda acceder a diferentes opciones de acuerdo a un rol.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Ingresa al sistema con su nombre usuario.</li> <li>• El sistema muestra las opciones de menú y campos de formularios de acuerdo al rol.</li> </ul>		

*Tabla 4 Historia de Usuario Autenticación de Usuarios*

Historia de Usuario			
<b>Número</b>	2	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Taxonomía.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Permitir consultar la taxonomía de una especie. Además, se podrán ingresar nuevos registros y actualizar los ya ingresados.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Ingresar una especie en la base de datos, seleccionando su taxonomía de manera jerárquica.</li> <li>• En caso de no tener la taxonomía completa, se ingresa los taxones faltantes.</li> <li>• Actualizar los datos de una especie.</li> <li>• Consultar la taxonomía de una especie.</li> <li>• Consultar la ficha de una especie.</li> </ul>		

*Tabla 5 Historia de Usuario Gestión de Taxonomía*

<b>Historia de Usuario</b>			
<b>Número</b>	3	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Contenido de las Paginas.		
<b>Prioridad en Negocio</b>	Media.		
<b>Descripción</b>	Mostrar información general del Herbario, similar a lo ya expuesto en puce.edu.ec. Permitir la edición del contenido de la página.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Presentar la información del herbario en secciones, similar a la página puce.edu.ec.</li> </ul>		

*Tabla 6 Historia de Usuario Gestión de Contenido de las Paginas*

<b>Historia de Usuario</b>			
<b>Número</b>	4	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Localidad.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Se requiere que el sistema permita el ingreso, consulta y actualización de localidades por medio un formulario que despliegue las divisiones geográficas jerárquicamente.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Ingresar una localidad en la base de datos.</li> <li>• En caso de no existir información completa de las divisiones geográficas superiores, se ingresa la división geográfica correspondiente.</li> <li>• Actualizar los datos de una localidad.</li> <li>• Consultar localidades.</li> <li>• Consultar detalles de una localidad.</li> </ul>		

*Tabla 7 Historia de Usuario Gestión de Localidad*

<b>Historia de Usuario</b>			
<b>Número</b>	5	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Espécimen.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Se requiere que el sistema permita el ingreso, consulta y actualización de especímenes por medio un formulario que despliegue la información necesaria para la identificación de la muestra.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Ingresar un espécimen asociado a una especie.</li> <li>• Actualizar los datos de un espécimen.</li> <li>• Consultar especímenes.</li> </ul>		

*Tabla 8 Historia de Usuario Gestión de Espécimen*

<b>Historia de Usuario</b>			
<b>Número</b>	6	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Publicaciones.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Se requiere que el sistema permita el ingreso, consulta y actualización de publicaciones científicas por medio un formulario, y la información a de su autor.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Ingresar una publicación y seleccionar autor.</li> <li>• Actualizar los datos de una publicación.</li> <li>• Ingresar autores.</li> <li>• Actualizar autores.</li> </ul>		

*Tabla 9 Historia de Usuario Gestión de Publicaciones*

Historia de Usuario			
<b>Número</b>	7	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Préstamos.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Se requiere que el sistema permita el registro, consulta y actualización de préstamos realizados por el personal del herbario por medio un formulario, y la información del beneficiario.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Registrar un préstamo y seleccionar el beneficiario.</li> <li>• Actualizar el estado del préstamo.</li> <li>• Ingresar beneficiarios.</li> <li>• Actualizar beneficiarios.</li> </ul>		

*Tabla 10 Historia de Usuario Gestión de Préstamos*

## 2.5 Recursos y Costos del Proyecto

Inicialmente, el proyecto estaba planeado para utilizar la infraestructura del Museo de Biología. Sin embargo, tras la decisión de unificar los proyectos de bases de datos que se encontraban en desarrollo, es decir, el Proyecto BioWeb Ecuador y el Sistema web para la administración del Herbario QCA, se acordó utilizar un solo sistema que utilice una base de datos que agrupe todas las áreas.

Se decide utilizar una plataforma de servicios de computación en la nube. Después de un análisis, enfocado principalmente a costos y la tecnología disponible actualmente, entre distintos proveedores, se acordó utilizar Microsoft Azure.

(Ver Anexo 2)

A continuación, se muestran los costos detallados para la contratación del servicio utilizando Microsoft Azure:

Servicio	Costo		
	Cantidad	Mensual	Anual
<b>Servicio de base de datos SQLServer</b> Tipo de instancia S2 Procesamiento: DTU 50 Tamaño de BDD: 250 GB	1	75	900
<b>Servicio de aplicaciones hosting BioWeb</b> Tipo B2 Core 2 RAM 3.5 GB	1	112	1344
<b>Servicio de aplicaciones hosting Web API</b> Tipo B2 Core 2 RAM 3.5 GB	1	112	1344
<b>Ancho de banda de salida</b> 10 GB	1	1	12
<b>Almacenamiento</b> 1 TB Tipo block Redundancia local	1	24	288
<b>Direcciones IP Reservadas</b> 5 IP Publicas	5	0	0
<b>Sub-Total:</b>		<b>324</b>	<b>3888</b>
<b>Variación estimada</b>	25%	81	972
<b>Total:</b>		<b>405</b>	<b>4860</b>

Tabla 11 Costos detallados Servicio Microsoft Azure

Los precios son referenciales en base a la información de Microsoft con la siguiente referencia:

<https://azure.microsoft.com/en-us/pricing/calculator/>

El software utilizado para el desarrollo de software será el siguiente:

<b>Cant.</b>	<b>Producto</b>	<b>Descripción</b>	<b>Precio</b>
2	MS SQL Express 2014	Sistema de Administración de Base de Datos.	0
2	Visual Studio 2015 Community Edition	Software para el desarrollo de aplicaciones para plataforma Windows.	0
1	Visual Studio Team Foundation Express 2012	Sistema de Control de Código Fuente.	0
1	Licencia de Office 2013.	La licencia se encuentra instaladas en el computador del personal de desarrollo de software.	0
1	Herramienta de inteligencia de negocios PowerBI, QlikView o Tableau	Las licencias pueden ser obtenidas en colaboración con la Facultad de Ingeniería de Sistemas dentro del programa DreamSpark de Microsoft.	0
		<b>Total USD:</b>	<b>0</b>

*Tabla 12 Software usado en el desarrollo del proyecto*

El tipo de licencias y cantidad de licencia de Office son administradas por la Dirección de Informática de la PUCE, entendiéndose que las que están actualmente instaladas fueron autorizadas en conformidad con la normativa de la universidad.

## 2.6 Diagramas de Caso de Uso

Una vez definidas las historias de usuario y la pila del producto (Pila del Producto), se identifican los actores del sistema y se realizan los diagramas de caso de uso.

Los diagramas de Caso de Uso proporcionan una visión de lo que el sistema debe hacer desde el punto de vista del usuario, no representan el diseño y no describen los elementos internos del sistema.

### 2.6.1 Diagrama de Casos de Uso General

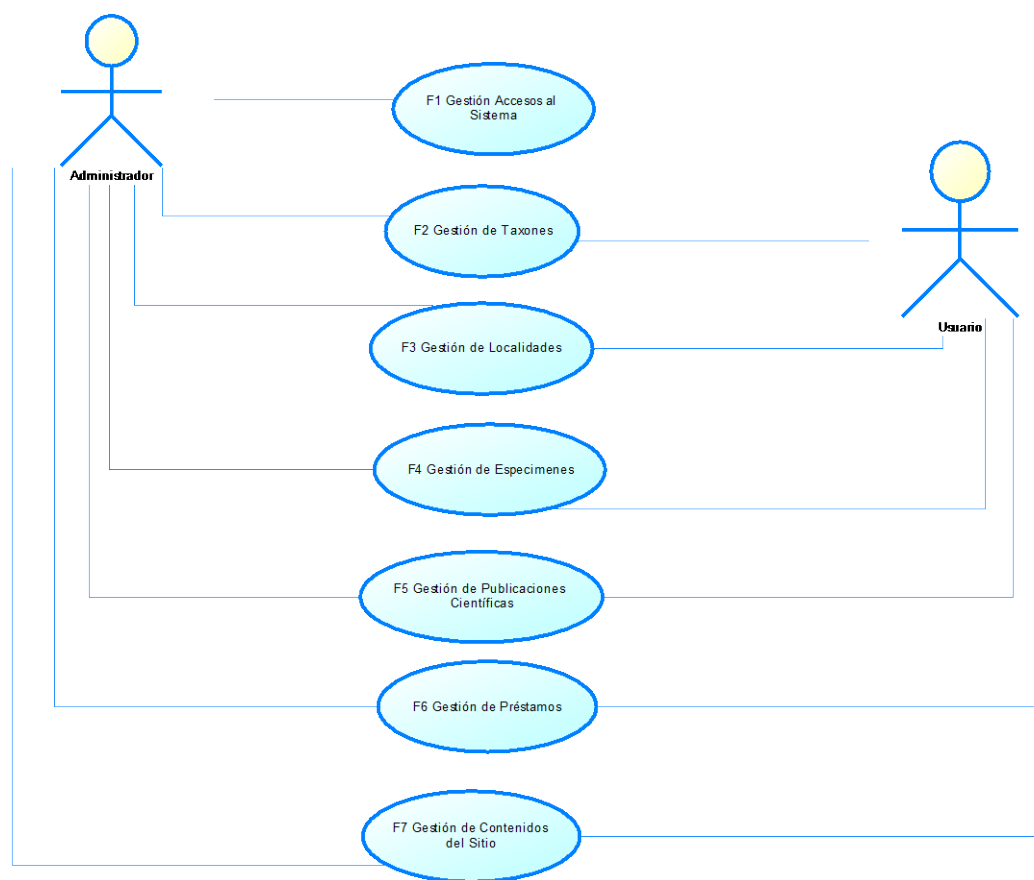


Diagrama 1 Casos de Uso General del Sistema

<b>Roles Actuales</b>	<b>Permisos</b>
Administrador	Control total del sistema.
Usuario	Los permisos dependen del rol asignado.

Tabla 13 Detalle Actores del Sistema

## 2.6.2 Diagrama F1: Gestión Accesos al Sistema

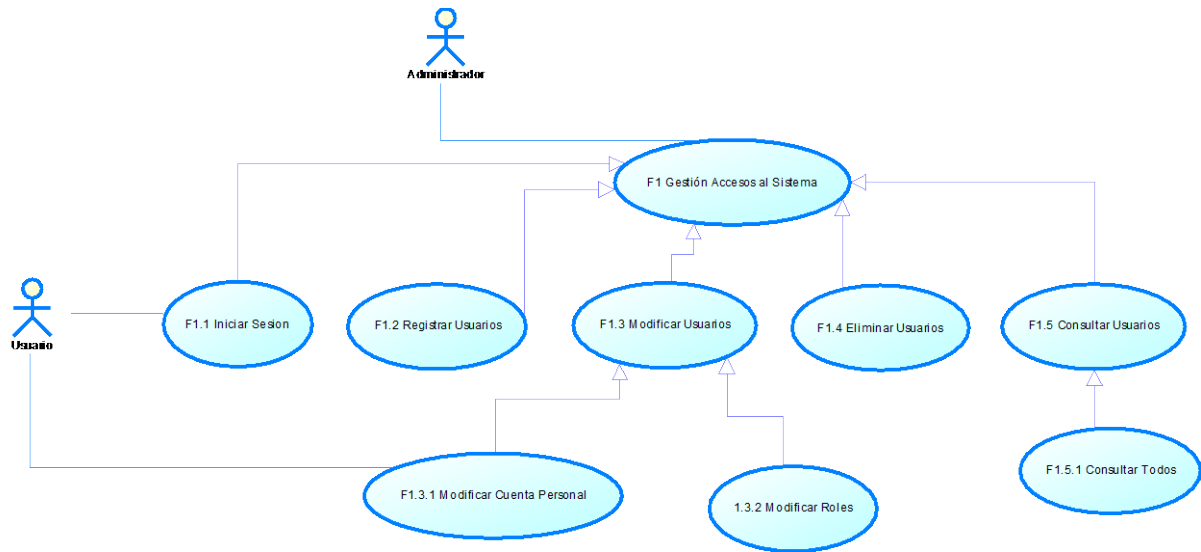
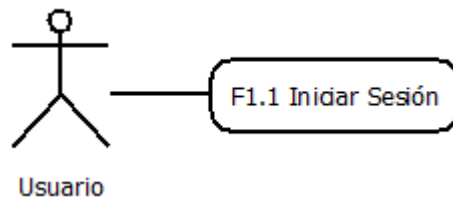


Diagrama 2 Caso de Uso Gestión Accesos al Sistema

### Caso de uso a detalle:

#### Caso de uso: F1.1: Iniciar Sesión

Actores: Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración. (E1)

### Flujo Alternativo:

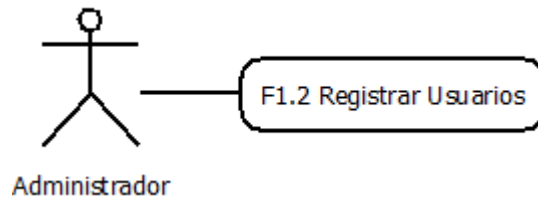
4. El sistema no encuentra el nombre de usuario, el administrador del sistema debe agregar al usuario al sistema.

**Excepciones:**

**E1:** El sistema no puede validar las credenciales, Intentar nuevamente o comunicarse con el administrador del sistema.

**Caso de uso: F1.2: Registrar Usuarios**

**Actores:** Administrador.

**Flujo Principal:**

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales de administrador y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la pestaña usuarios y expande su contenido.
6. El sistema presenta el menú de registro de usuarios.
7. El actor ingresa los datos y rol del usuario y presiona el botón guardar.
8. El sistema almacena la información. (E1)

**Flujo Alternativo:**

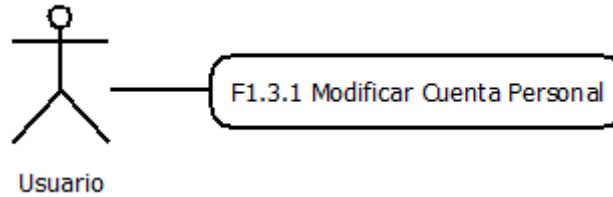
7. No existe el rol deseado, el actor presiona el botón para agregar roles e ingresa el rol deseado.

**Excepciones:**

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F1.3.1: Modificar Cuenta Personal

Actores: Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor presiona el botón 'Cuenta' ubicado en la sección superior derecha.
6. El sistema presenta la sección de cuenta de usuario.
7. El actor modifica sus datos personales y presiona el botón guardar.
8. El sistema almacena la información. (E1)

### Flujo Alternativo:

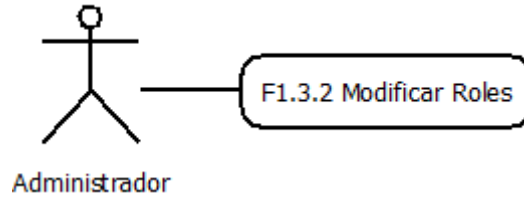
4. El sistema no encuentra el nombre de usuario, el administrador del sistema debe agregar al usuario al sistema.

### Excepciones:

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F1.3.2: Modificar Roles

Actores: Administrador.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales de administrador y presiona el Botón Iniciar sesión
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Usuarios.
6. El sistema presenta el menú de Modificación de usuarios.
7. El actor selecciona La pestaña usuarios.
8. El sistema muestra los usuarios.
9. El actor selecciona el usuario, modifica el rol asignado y pulsa el botón guardar.
10. El sistema almacena la información. (E1)

### Flujo Alternativo:

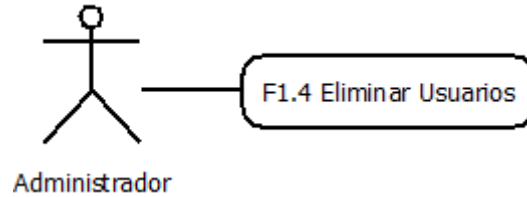
9. No existe el rol deseado, el actor presiona el botón para agregar roles e ingresa el rol deseado.

### Excepciones:

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F1.4: Eliminar Usuarios

Actores: Administrador.



### Flujo Principal:

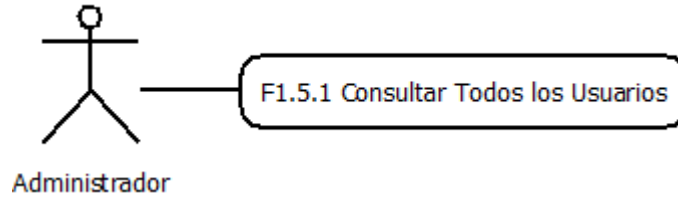
1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales de administrador y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Usuarios.
6. El sistema presenta el menú de Modificación de usuarios.
7. El actor selecciona La pestaña usuarios.
8. El sistema muestra los usuarios.
9. El actor selecciona el usuario, Y presiona el botón Eliminar.
10. El sistema elimina el usuario escogido. (E1)

### Excepciones:

**E1:** El sistema no puede eliminar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F1.5.1 Consultar Todos los Usuarios

Actores: Administrador.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales de administrador y presiona el Botón Iniciar sesión
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Usuarios.
6. El sistema presenta el menú de Modificación de usuarios.
7. El actor selecciona La pestaña usuarios.
8. El sistema muestra los usuarios. (E1)
9. El actor filtra los resultados de la búsqueda ingresando información en el campo de búsqueda.
10. El sistema presenta la información filtrada. (E1)

### Excepciones:

**E1:** El sistema no presenta la información, Intentar nuevamente o comunicarse con el DBA.

### 2.6.3 Diagrama F2: Gestión de Taxones

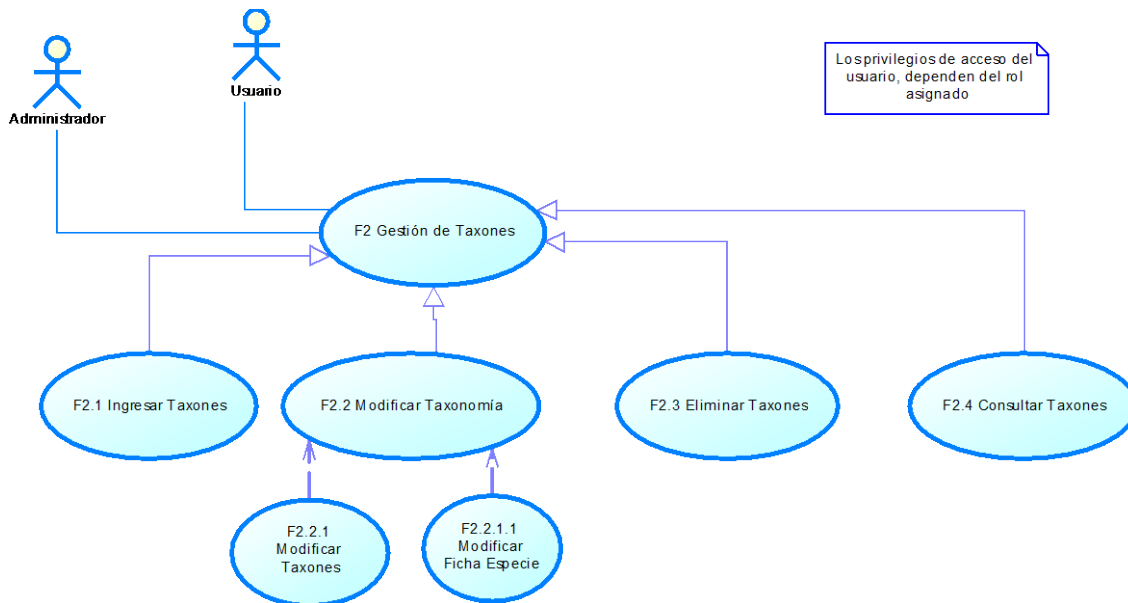
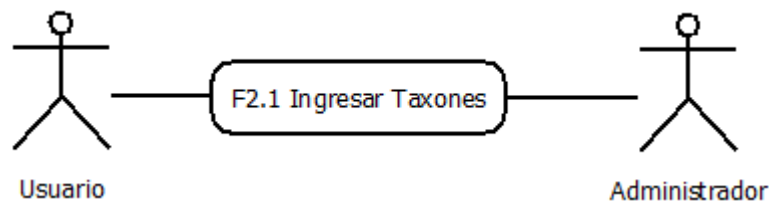


Diagrama 3 Caso de Uso Gestión de Taxones

#### Caso de uso: F2.1: Ingresar Taxones

Actores: Administrador, Usuario.



#### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Taxonomía.
6. El sistema presenta el Menú de Taxonomía.

7. El actor Selecciona la categoría de taxón que desea ingresar (Reino, Filo, Clase, Orden, Familia, Especie).
8. El sistema muestra el formulario de ingreso de taxón.
9. El usuario ingresa los datos correspondientes y presiona el botón Guardar.
10. El sistema almacena la información. (E1)

**Flujo Alternativo:**

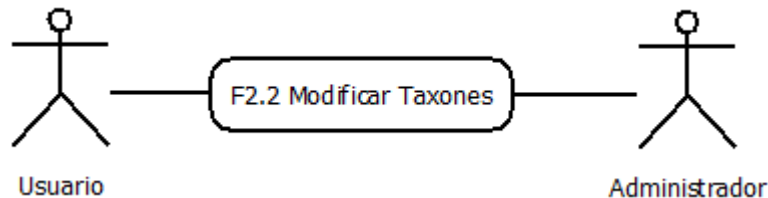
7. El usuario debe agregar primero un taxón padre antes de agregar un taxón hijo, EJ. Para agregar un Filo primero debe agregar un Reino.

**Excepciones:**

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F2.2.1: Modificar Taxones

Actores: Administrador, Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Taxonomía.
6. El sistema presenta el Menú de Taxonomía.
7. El usuario Ingresa el nombre y categoría del taxón que desea buscar en el campo de búsqueda, da click en buscar y click en el ítem deseado de los resultados.
8. El sistema muestra la ficha del taxón.
9. El usuario modifica los datos correspondientes y presiona el botón Guardar.
10. El sistema almacena la información. (E1)

### Flujo Alternativo:

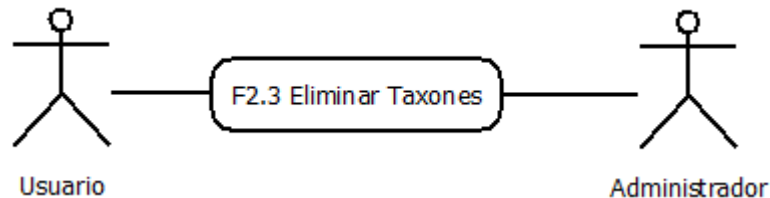
7. El usuario debe agregar primero un taxón padre antes de agregar un taxón hijo, EJ. Para agregar un Filo primero debe agregar un Reino.

### Excepciones:

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F2.3: Eliminar Taxones

Actores: Administrador, Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Taxonomía.
6. El sistema presenta el Menú de Taxonomía.
7. El usuario Ingresa el nombre y categoría del taxón que desea buscar en el campo de búsqueda, da click en buscar y click en el ítem deseado de los resultados.
8. El sistema muestra la ficha del taxón.
9. El usuario verifica el taxón y presiona el botón Eliminar.
10. El sistema almacena la información. (E1)

### Flujo Alternativo:

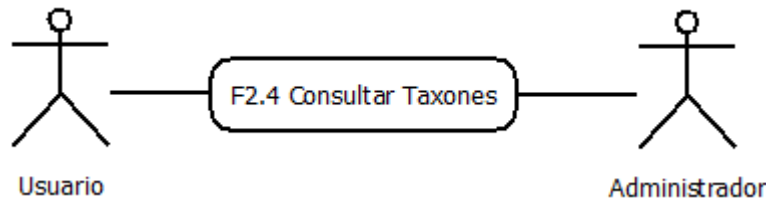
9. El sistema muestra un error de dependencias, El usuario primero debe eliminar todas las relaciones de dicho taxón, EJ. Para eliminar un Género primero debe eliminar la Especie asociada a ese género y los especímenes de dicha Especie.

### Excepciones:

E1: El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F2.4: Consultar Taxones

Actores: Administrador, Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Taxonomía.
6. El sistema presenta el Menú de Taxonomía.
7. El usuario Ingresa el nombre y categoría del taxón que desea buscar en el campo de búsqueda, da click en buscar.
8. El sistema muestra los taxones que cumplen con el criterio de búsqueda. (E1)

### Flujo Alternativo:

7. El usuario puede navegar independientemente por las pestañas de cada uno de los grupos taxonómicos (Reino, Filo, Clase, Orden, Familia, Especie).
8. El sistema no muestra información, no existen taxones que cumplan con el criterio de búsqueda.

### Excepciones:

E1: El sistema muestra un error de consulta, Intente nuevamente o comuníquese con el DBA.

## 2.6.4 Diagrama F3: Gestión de Localidades

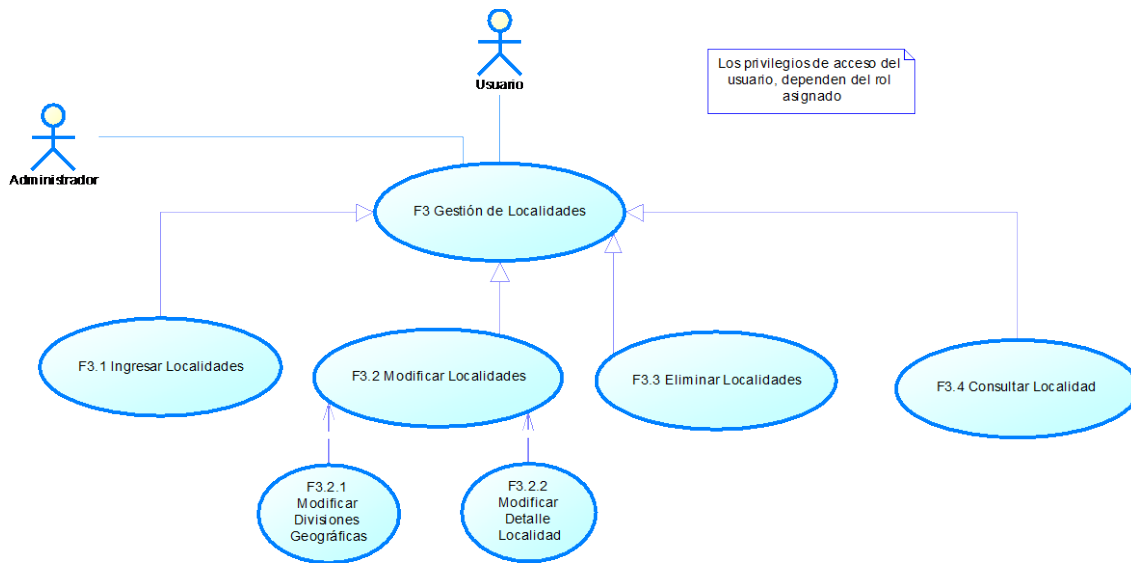
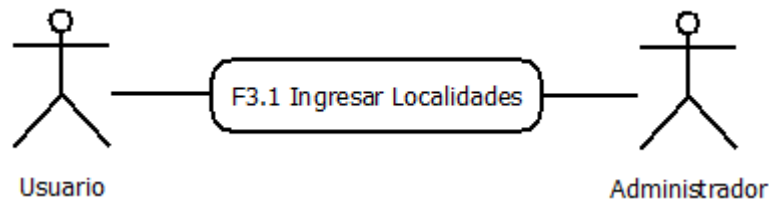


Diagrama 4 Caso de Uso Gestión de Localidades

### Caso de uso: F3.1: Ingresar Localidades

Actores: Administrador, Usuario.



#### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección de Localidades.
6. El sistema presenta el Menú de Localidades.
7. El actor Selecciona la categoría de localidad que desea ingresar (Continente, País, Provincia, Cantón, Parroquia, Localidad).

8. El sistema muestra el formulario de ingreso de Localidad.
9. El usuario ingresa los datos correspondientes y presiona el botón Guardar.
10. El sistema almacena la información. (E1)

**Flujo Alternativo:**

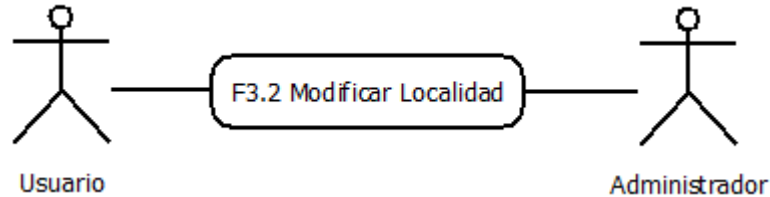
7. El usuario debe agregar primero una categoría superior de localidad, EJ. Para agregar una provincia, primero debe existir un país al cual asignar la nueva provincia.

**Excepciones:**

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F3.2: Modificar Localidad

Actores: Administrador, Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Localidad.
6. El sistema presenta el Menú de Localidad.
7. El usuario Ingresa el nombre y tipo de localidad que desea buscar en el campo de búsqueda da click en buscar y da click en el ítem deseado.
8. El sistema muestra la ficha de localidad.
9. El usuario modifica los datos correspondientes y presiona el botón Guardar.
10. El sistema almacena la información. (E1)

### Flujo Alternativo:

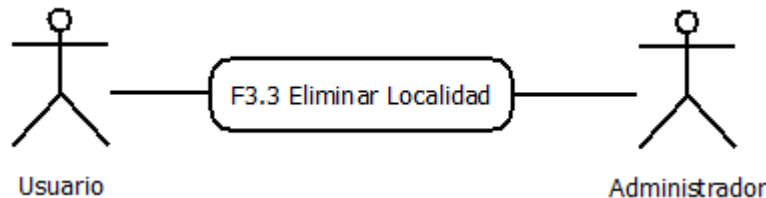
9. El usuario debe agregar primero una categoría superior de localidad, EJ. Para agregar una provincia, primero debe existir un país al cual asignar la nueva provincia.

### Excepciones:

E1: El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

### Caso de uso: F3.3: Eliminar Localidad

Actores: Administrador, Usuario.



#### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Localidad.
6. El sistema presenta el Menú de Localidad.
7. El usuario Ingresa el nombre y tipo de localidad que desea buscar en el campo de búsqueda da clic en buscar y da clic en el ítem deseado.
8. El sistema muestra la ficha de localidad.
9. El usuario verifica los datos correspondientes y presiona el botón Eliminar.
10. El sistema elimina la información. (E1)

#### Flujo Alternativo:

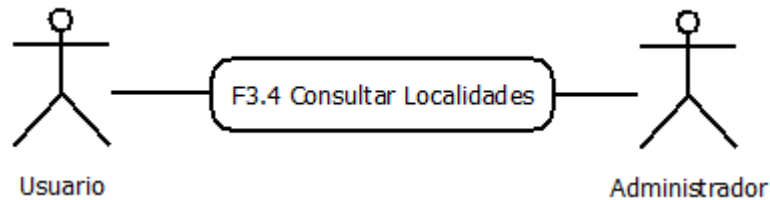
9. El sistema muestra un error de dependencias, El usuario primero debe eliminar todas las relaciones de dicha localidad, EJ. Para eliminar un Cantón, primero debe eliminar las Parroquias asociada a ese Cantón y las localidades propias de dicha Parroquia.

#### Excepciones:

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F3.4: Consultar Localidades

Actores: Administrador, Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Localidad.
6. El sistema presenta el Menú de Localidad.
7. El usuario Ingresa el nombre y categoría de la localidad que desea buscar en el campo de búsqueda, da click en buscar.
8. El sistema muestra las localidades que cumplen con el criterio de búsqueda. (E1)

### Flujo Alternativo:

7. El usuario puede navegar independientemente por las pestañas de cada uno de los grupos de Localidades (Continente, País, Provincia, Cantón, Parroquia, Localidad).
8. El sistema no muestra información, no existen localidades que cumplan con el criterio de búsqueda.

### Excepciones:

E1: El sistema muestra un error de consulta, Intente nuevamente o comuníquese con el DBA.

## 2.6.5 Diagrama F4: Gestión de Especímenes

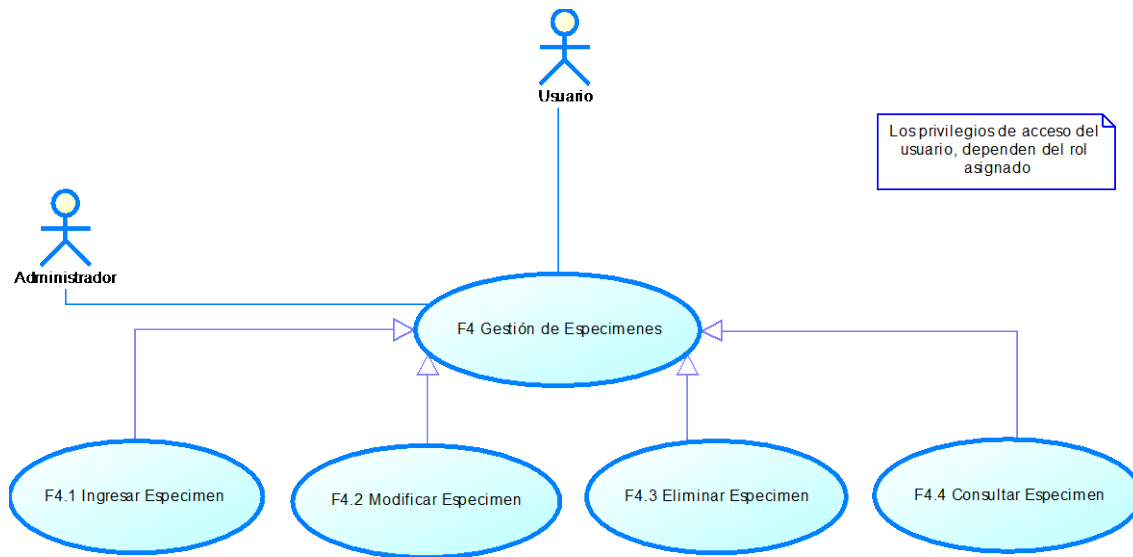
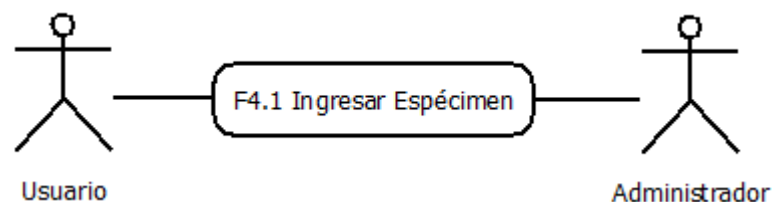


Diagrama 5 Caso de Uso Gestión de Especímenes

### Caso de uso: F4.1: Ingresar Espécimen

**Actores:** Administrador, Usuario



#### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección de Especímenes en taxonomía.
6. El sistema presenta el Menú de Especímenes.
7. El actor Selecciona la taxonomía superior a la cual asociar el espécimen y da click en añadir Especimen.

8. El sistema muestra el formulario de ingreso de Espécimen.
9. El usuario ingresa los datos correspondientes y presiona el botón Guardar.
10. El sistema almacena la información. (E1)

**Flujo Alternativo:**

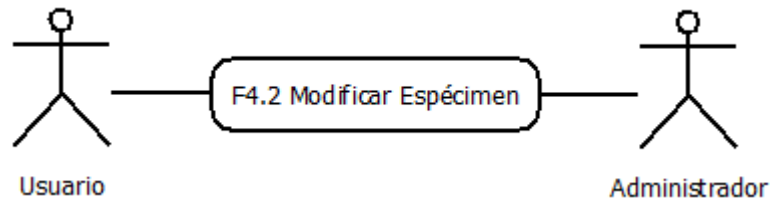
7. No existe la especie necesaria, el usuario debe añadir la especie requerida desde el menú Taxonomía o Ficha.

**Excepciones:**

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F4.2: Modificar Espécimen

Actores: Administrador, Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Especímenes.
6. El sistema presenta el Menú de Especímenes.
7. El usuario Ingresa el código del espécimen en el campo de búsqueda y da click en el botón Buscar.
8. El sistema muestra los datos encontrados.
9. El usuario da click en el resultado deseado.
10. El sistema muestra la ficha del Especímen.
11. El usuario modifica los datos correspondientes y presiona el botón Guardar.
12. El sistema almacena la información. (E1).

### Flujo Alternativo:

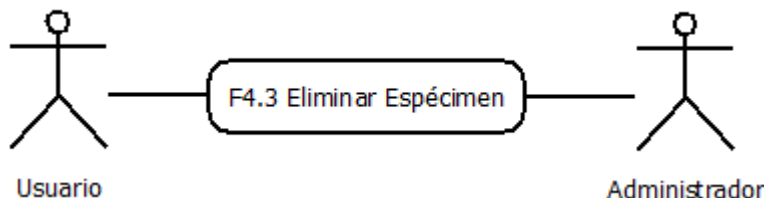
7. El usuario busca el Especímen navegando por la taxonomía superior a la cual pertenece dicho espécimen.

### Excepciones:

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

### Caso de uso: F4.3: Eliminar Espécimen.

Actores: Administrador, Usuario.



#### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Especímenes.
6. El sistema presenta el Menú de Especímenes.
7. El usuario Ingresa el código del espécimen en el campo de búsqueda y da click en el botón Buscar.
8. El sistema muestra los datos encontrados.
9. El usuario da click en el resultado deseado.
10. El sistema muestra la ficha del Especímen.
11. El usuario verifica los datos correspondientes y presiona el botón Eliminar.
12. El sistema elimina la información. (E1).

#### Flujo Alternativo:

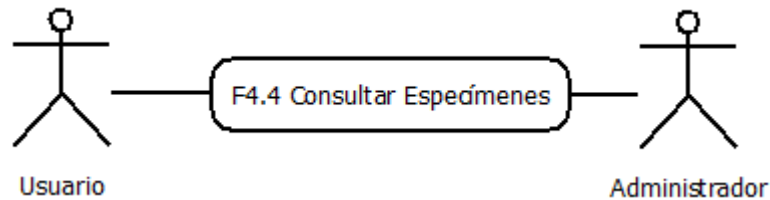
7. El usuario busca el Especímen navegando por la taxonomía superior a la cual pertenece dicho espécimen.
7. El usuario no ingresa ningún dato en el campo de búsqueda y presiona el botón Ver Todos.

#### Excepciones:

**E1:** El sistema no puede guardar la información, Intentar nuevamente o comunicarse con el DBA.

## Caso de uso: F4.4: Consultar Especímenes

Actores: Administrador, Usuario.



### Flujo Principal:

1. El actor ingresa a la página web de administración.
2. El sistema muestra la página web de Inicio de Sesión.
3. El actor Ingresa sus credenciales y presiona el Botón Iniciar sesión.
4. El sistema presenta el sitio web de administración.
5. El actor se dirige a la sección Especímenes.
6. El sistema presenta el Menú de Especímenes.
7. El usuario Ingresa el código del espécimen en el campo de búsqueda y da click en el botón Buscar.
8. El sistema muestra los datos encontrados. (E1)

### Flujo Alternativo:

7. El usuario busca el Espécimen navegando por la ficha de la Especie a la cual pertenece dicho espécimen.
8. El sistema no muestra información, no existen localidades que cumplan con el criterio de búsqueda.

### Excepciones:

E1: El sistema muestra un error de consulta, Intente nuevamente o comuníquese con el DBA.

## 2.6.6 Diagrama F5: Gestión de Publicaciones Científicas

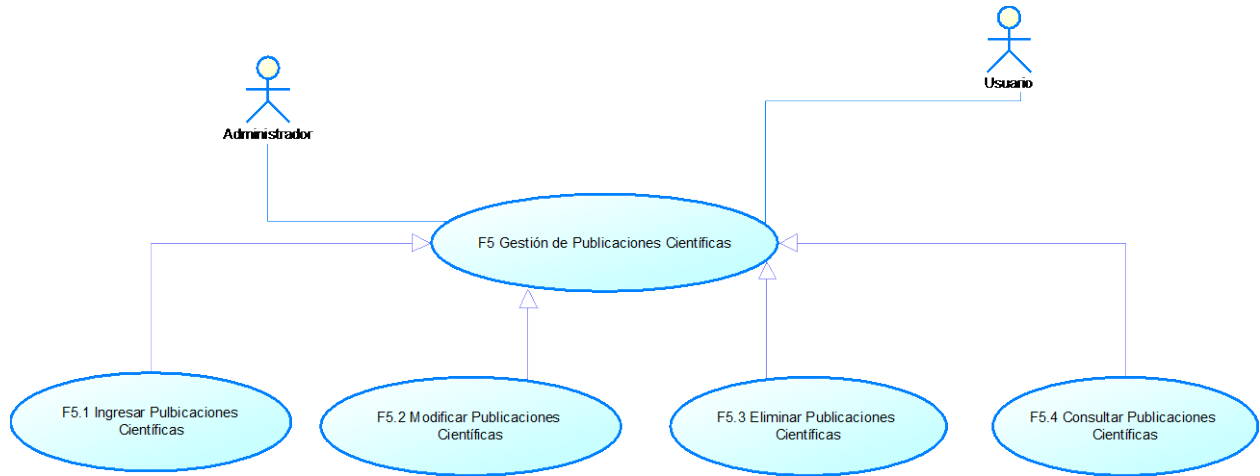
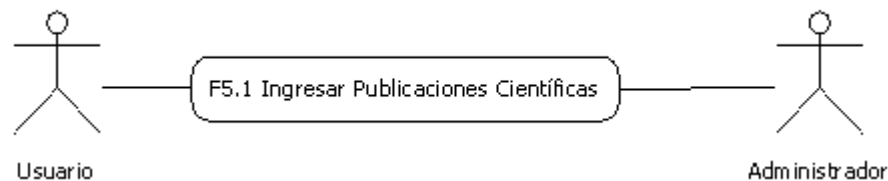


Diagrama 6 Caso de Uso Gestión de Publicaciones Científicas

**Caso de Uso:** F5.1 Ingresar Publicaciones Científicas.

**Descripción:** El usuario del sistema podrá realizar ingresos de publicaciones Científicas relacionadas a un taxón.

**Actores:** Administrador y Usuario registrado.



### Flujo Principal

- 1.- El actor selecciona la opción Publicaciones en el menú.
- 2.- El sistema muestra la página Publicaciones y despliega el acordeón Datos de Publicación.
- 3.- El actor selecciona la opción Nueva Publicación.
- 4.- El sistema muestra la página Nueva Publicación.
- 5.- El actor ingresa los datos de la Publicación.
- 6.- El actor presiona el botón Guardar Publicación. (E1, E2, E3)

- 7.- El sistema guarda los datos de la Publicación.
- 8.-El sistema despliega el acordeón Autor(es).
- 9.- El actor Selecciona el Autor de la Publicación.
- 10.- El actor presiona el botón Añadir Autor. (E1, E2, E3)
- 11.- El sistema asocia la publicación al autor seleccionado.
- 12.- El sistema despliega el acordeón Archivo.
- 13.- El actor selecciona un archivo.
- 14.- El actor presiona el botón Subir Archivo. (E2, E3)
- 15.- El sistema sube el archivo y lo asocia con la publicación.

#### **Flujo Alternativo**

- 9.- Si no existe autor, el actor debe Ingresar los datos relacionados con autor.

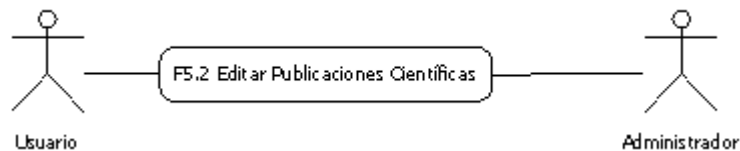
#### **Excepciones**

- E1:** Problema de ingreso de datos, no se ingresaron los datos obligatorios.
- E2:** Existe un problema con la conexión a la base de datos.
- E3:** No existe conexión a Internet.

**Caso de Uso:** F5.2 Editar Publicaciones Científicas.

**Descripción:** El usuario del sistema podrá realizar modificaciones a las publicaciones Científicas registradas.

**Actores:** Administrador y Usuario registrado.



### Flujo Principal

- 1.- El actor selecciona la opción Publicaciones en el menú.
- 2.- El sistema muestra la página Publicaciones y despliega el acordeón Datos de Publicación.
- 3.- El actor busca la publicación filtrando por autor, título, año, editorial.
- 4.- El sistema muestra una lista de publicaciones de la búsqueda.
- 5.- El actor selecciona la publicación y presiona el botón Editar.
- 6.- El sistema muestra la página Editar Publicación con los datos de la misma.
- 7.- El actor modifica los datos de la Publicación.
- 8.- El actor presiona el botón Guardar Publicación. (E1, E2, E3)
- 9.- El sistema actualiza los datos de la Publicación.
- 10.-El sistema despliega el acordeón Autor(es).
- 11.- El actor Selecciona el Autor de la Publicación.
- 12.- El actor presiona el botón Añadir Autor. (E1, E2, E3)
- 13.- El sistema actualiza la publicación al autor seleccionado.
- 14.- El sistema despliega el acordeón Archivo.
- 15.- El actor selecciona un archivo.
- 16.- El actor presiona el botón Subir Archivo. (E2, E3)
- 17.- El sistema actualiza o sube un nuevo archivo y lo asocia con la publicación.

### **Flujo Alterno**

11.- Si no existe autor, el actor debe Ingresar los datos relacionados con autor.

### **Excepciones**

**E1:** Problema al actualizar los datos, no se ingresaron los datos obligatorios.

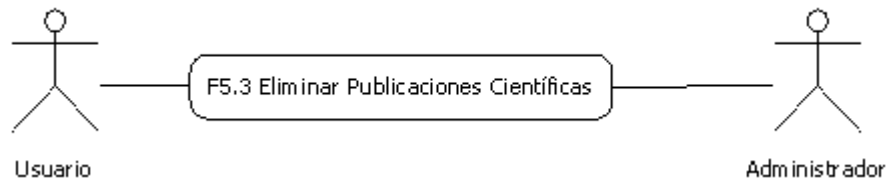
**E2:** Existe un problema con la conexión a la base de datos.

**E3:** No existe conexión a Internet.

**Caso de Uso:** F5.3 Eliminar Publicaciones Científicas.

**Descripción:** El usuario del sistema podrá eliminar a las publicaciones Científicas registradas.

**Actores:** Administrador y Usuario registrado.



### **Flujo Principal**

- 1.- El actor selecciona la opción Publicaciones en el menú.
- 2.- El sistema muestra la página Publicaciones y despliega el acordeón Datos de Publicación.
- 3.- El actor busca la publicación filtrando por autor, título, año, editorial.
- 4.- El sistema muestra una lista de publicaciones de la búsqueda.
- 5.- El actor selecciona la publicación y presiona el botón Eliminar.
- 6.- El sistema pide confirmación para eliminar la publicación.
- 7.- El sistema confirma la eliminación de la publicación.
- 8.- El sistema elimina la publicación.

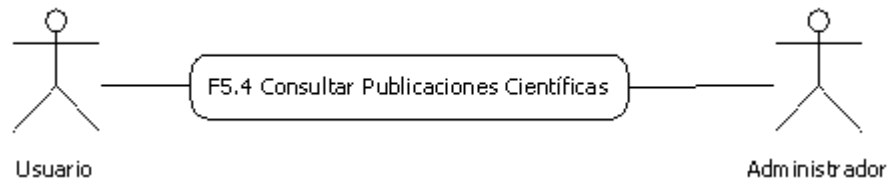
### **Excepciones**

- E1:** Problema al eliminar la publicación, la base de datos.
- E2:** Existe un problema con la conexión a la base de datos.
- E3:** No existe conexión a Internet.

**Caso de Uso:** F5.4 Consultar Publicaciones Científicas.

**Descripción:** El usuario del sistema podrá realizar modificaciones a las publicaciones Científicas registradas.

**Actores:** Administrador y Usuario registrado.



### Flujo Principal

- 1.- El actor selecciona la opción Publicaciones en el menú.
- 2.- El sistema muestra la página Publicaciones y despliega el acordeón Datos de Publicación.
- 3.- El actor busca la publicación filtrando por autor, título, año, editorial.
- 4.- El sistema muestra una lista de publicaciones de la búsqueda. (E1, E2, E3)
- 5.- El actor selecciona la publicación y presiona el botón Visualizar.
- 6.- El sistema muestra la página Ver Publicación con los datos de la misma.

### Excepciones

- E1:** No se encuentran los datos solicitados.
- E2:** Existe un problema con la conexión a la base de datos.
- E3:** No existe conexión a Internet.

## 2.6.7 Diagrama F6: Gestión de Préstamos

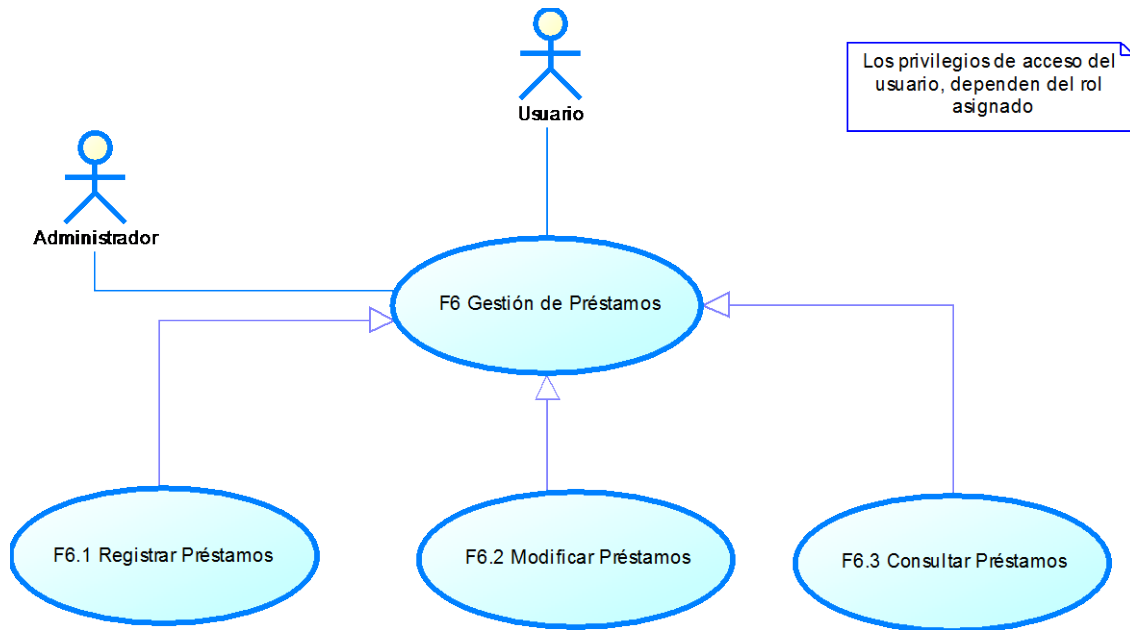


Diagrama 7 Caso de Uso Gestión de Préstamos

**Caso de Uso:** F6.1 Registrar Préstamos.

**Descripción:** El usuario del sistema podrá registrar los préstamos que se realicen.

**Actores:** Administrador y Usuario registrado.



### Flujo Principal

- 1.- El actor selecciona la opción Préstamos en el menú.
- 2.- El sistema muestra la página Préstamos y despliega el acordeón Datos de Préstamo.
- 3.- El actor selecciona la opción Nuevo Préstamo.
- 4.- El sistema muestra la página Nuevo Préstamo.
- 5.- El actor ingresa los datos del Préstamo.
- 6.- El actor presiona el botón Guardar Préstamo. (E1, E2, E3)

7.- El sistema guarda los datos del Préstamo.

**Excepciones**

**E1:** Problema de ingreso de datos, no se ingresaron los datos obligatorios.

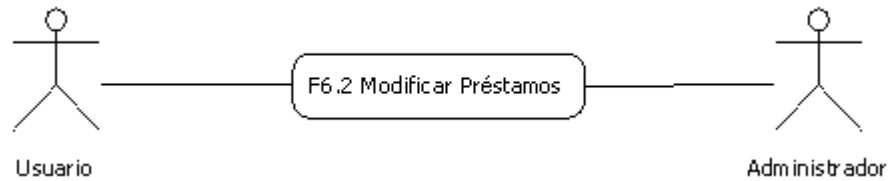
**E2:** Existe un problema con la conexión a la base de datos.

**E3:** No existe conexión a Internet.

**Caso de Uso:** F6.2 Modificación de Préstamos.

**Descripción:** El usuario del sistema podrá modificar datos del préstamo, entre estos: Datos del beneficiario, tiempo de entrega, ítem prestado, lugar de destino, emisor del préstamo, responsable de autorización.

**Actores:** Administrador y Usuario registrado.



### Flujo Principal

- 1.- El actor selecciona la opción Préstamos en el menú.
- 2.- El sistema muestra la página Préstamo y despliega el acordeón Datos de Préstamo.
- 3.- El actor busca el préstamo filtrando por personal, código, fecha.
- 4.- El sistema muestra una lista de préstamos de la búsqueda.
- 5.- El actor selecciona el préstamo y presiona el botón Editar.
- 6.- El sistema muestra la página Editar Préstamos con los datos de la misma.
- 7.- El actor modifica los datos del Préstamo.
- 8.- El actor presiona el botón Guardar Préstamo. (E1, E2, E3)
- 9.- El sistema actualiza los datos del Préstamo.

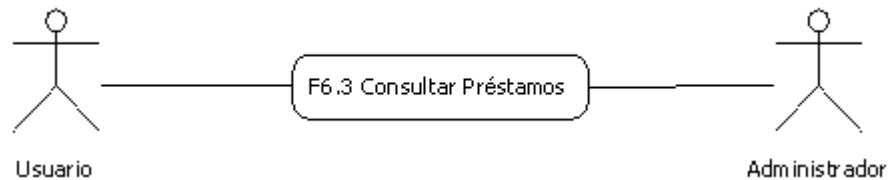
### Excepciones

- E1:** Problema al actualizar los datos, no se ingresaron los datos obligatorios.
- E2:** Existe un problema con la conexión a la base de datos.
- E3:** No existe conexión a Internet.

**Caso de Uso:** F6.3 Consultar Préstamos.

**Descripción:** El usuario del sistema podrá realizar modificaciones a las publicaciones Científicas registradas.

**Actores:** Administrador y Usuario registrado.



### Flujo Principal

- 1.- El actor selecciona la opción Préstamos en el menú.
- 2.- El sistema muestra la página Publicaciones y despliega el acordeón Datos de Publicación.
- 3.- El actor busca el préstamo filtrando por personal, código, fecha.
- 4.- El sistema muestra una lista de préstamos referentes a la búsqueda. (E1, E2, E3)
- 5.- El actor selecciona el Préstamo y presiona el botón Visualizar.
- 6.- El sistema muestra la página Ver Préstamo con los datos de la misma.

### Excepciones

- E1:** No se encuentran los datos solicitados.
- E2:** Existe un problema con la conexión a la base de datos.
- E3:** No existe conexión a Internet.

## 2.6.8 Diagrama F7: Gestión de Contenidos del Sitio

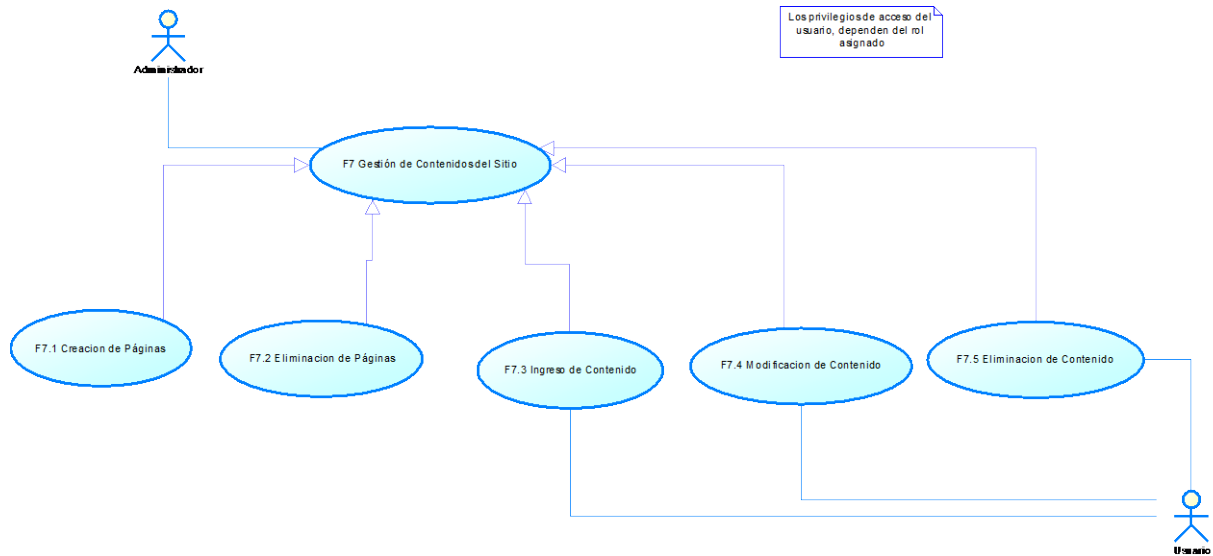


Diagrama 8 Caso de Uso Gestión de Contenidos del Sitio

### Caso de Uso: F7.1 Creación de Páginas.

**Descripción:** El Administrador del sistema tendrá la capacidad de agregar páginas y secciones adicionales al sitio, usando la interfaz de administración.

**Actores:** Administrador.



### Flujo Principal

- 1.- El actor selecciona la opción Páginas en el menú.
- 2.- El sistema muestra la página correspondiente a Páginas y una lista de páginas disponibles.
- 3.- El actor selecciona la opción Nueva Página.
- 4.- El sistema solicita asignar un nombre a la nueva página.
- 5.- El usuario asigna un nombre a la nueva página.
- 6.- El usuario presiona el botón crear página.

**6.-** El sistema crea una plantilla para agregar contenido. (E1, E2, E3, E4)

**Excepciones**

**E1:** Problema de ingreso de datos, no se ingresaron los datos obligatorios.

**E2:** Existe un problema con la conexión a la base de datos.

**E3:** No existe conexión a Internet.

**E4:** Problema al crear la plantilla, no se encuentra el archivo.

**Caso de Uso:** F7.2 Eliminación de Páginas.

**Descripción:** El Administrador del sistema tendrá la capacidad de agregar páginas y secciones adicionales al sitio, usando la interfaz de administración.

**Actores:** Administrador.



### Flujo Principal

- 1.- El actor selecciona la opción Páginas en el menú.
- 2.- El sistema muestra la página correspondiente a Páginas y una lista de páginas disponibles.
- 3.- El actor selecciona una de las páginas de la lista.
- 4.- El usuario selecciona la opción eliminar página.
- 5.- El sistema solicita una confirmación para eliminar la página.
- 6.- El usuario presiona el botón aceptar.
- 6.- El sistema elimina la página y sus datos asociados. (E1, E2, E3)

### Excepciones

**E1:** Existe un problema con la conexión a la base de datos.

**E2:** No existe conexión a Internet.

**E3:** Problema al eliminar la plantilla, no se encuentra el archivo.

**Caso de Uso:** F7.3 Ingreso de Contenido.

**Descripción:** El usuario registrado, tendrá la capacidad de aumentar contenido (publicaciones) que será visualizado en las páginas del sitio web de presentación.

**Actores:** Administrador y Usuario registrado.



### **Flujo Principal**

- 1.- El actor selecciona la opción Páginas en el menú.
- 2.- El sistema muestra la página correspondiente a Páginas y una lista de páginas disponibles.
- 3.- El actor selecciona una de las páginas de la lista.
- 4.- El usuario selecciona la opción modificar contenido.
- 5.- El sistema muestra la página Ingresar Contenidos.
- 6.- El usuario ingresa el contenido de la página sea texto, imágenes o videos.
- 7.- El usuario presiona el botón Guardar Contenido.
- 7.- El sistema guarda la página y con el contenido asociado. (E1, E2, E3)

### **Excepciones**

**E1:** Problema al actualizar los datos, no se ingresaron los datos obligatorios.

**E2:** Existe un problema con la conexión a la base de datos.

**E3:** No existe conexión a Internet.

**Caso de Uso:** F7.4 Modificación de Contenido.

**Descripción:** El usuario registrado, tendrá la capacidad de editar el contenido (publicaciones), de forma dinámica, que será visualizado en las páginas del sitio web de presentación.

**Actores:** Administrador y Usuario registrado.



### **Flujo Principal**

- 1.- El actor selecciona la opción Páginas en el menú.
- 2.- El sistema muestra la página correspondiente a Páginas y una lista de páginas disponibles.
- 3.- El actor selecciona una de las páginas de la lista.
- 4.- El usuario selecciona la opción modificar contenido.
- 5.- El sistema muestra la página Editar Contenidos.
- 6.- El usuario ingresa el contenido de la página sea texto, imágenes o videos.
- 7.- El usuario presiona el botón Guardar Contenido.
- 8.- El sistema guarda la página y con el contenido asociado. (E1, E2, E3)

### **Excepciones**

- E1:** Problema al actualizar los datos, no se ingresaron los datos obligatorios.
- E2:** Existe un problema con la conexión a la base de datos.
- E3:** No existe conexión a Internet.

**Caso de Uso:** F7.5 Eliminación de Contenido.

**Descripción:** El usuario registrado, tendrá la capacidad de eliminar el contenido (publicaciones), de forma dinámica, de las páginas del sitio web de presentación.

**Actores:** Administrador y Usuario registrado.



### **Flujo Principal**

- 1.- El actor selecciona la opción Páginas en el menú.
- 2.- El sistema muestra la página correspondiente a Páginas y una lista de páginas disponibles.
- 3.- El actor selecciona una de las páginas de la lista.
- 4.- El usuario selecciona la opción eliminar contenido.
- 5.- El sistema solicita confirmación para eliminar contenido.
- 6.- El usuario presiona el botón aceptar.
- 7.- El sistema elimina el contenido de la página. (E1, E2, E3)

### **Excepciones**

**E1:** Problema al actualizar los datos, no se ingresaron los datos obligatorios.

**E2:** Existe un problema con la conexión a la base de datos.

**E3:** No existe conexión a Internet.

## 2.7 Diagramas de Actividades

### Diagrama de Actividades de Proceso Recolección y Almacenamiento de Especímenes

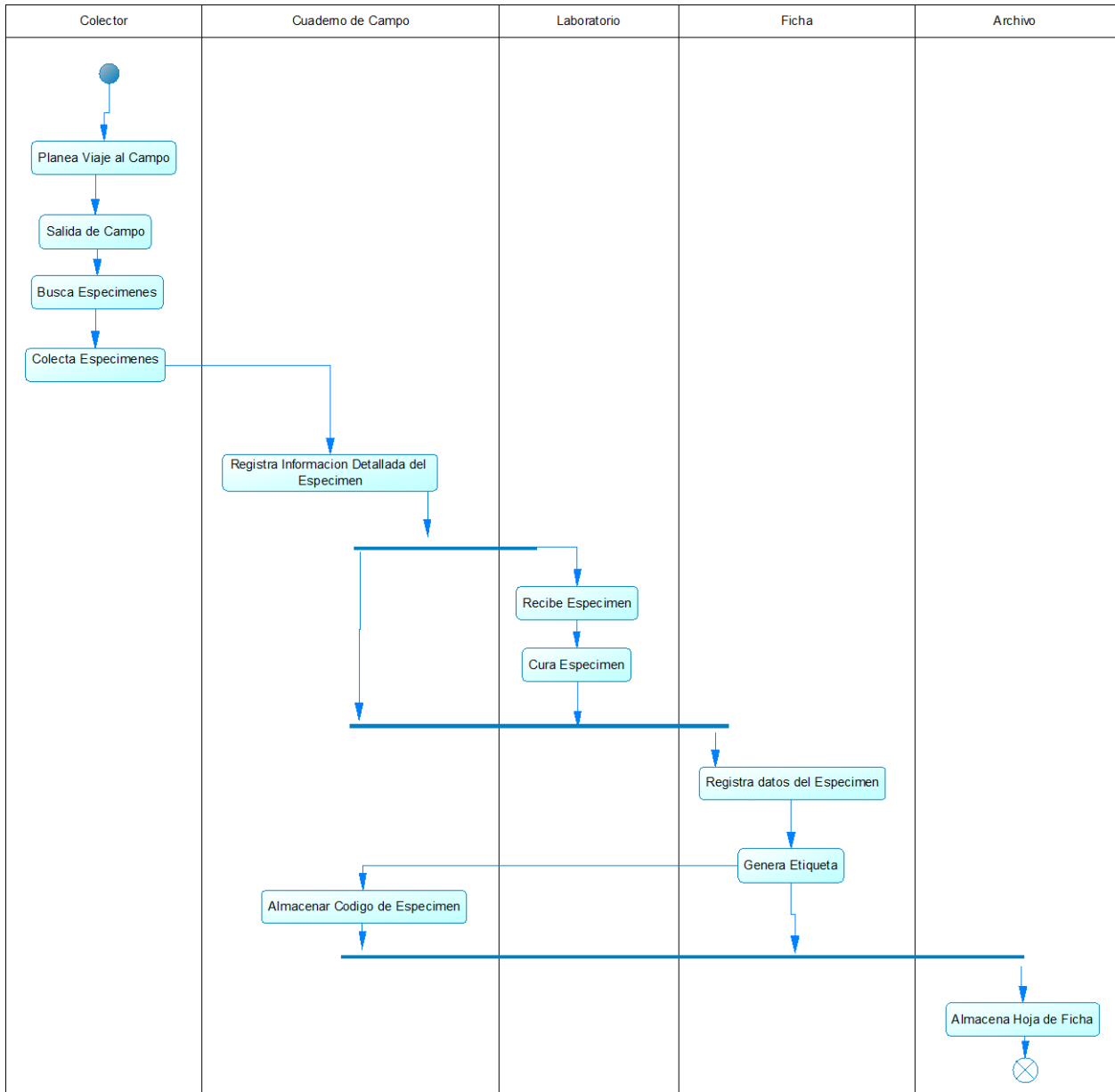


Diagrama 9 Actividades de Proceso Recolección y Almacenamiento de Especímenes

## 2.8 Estándares de Codificación

El sistema global se divide en 2 partes, El BackEnd y el FrontEnd, ambos tienen una arquitectura MVC (Modelo Vista Controlador), el BackEnd involucra la programación sobre la WebAPI de ASP.NET y las clases básicas del sistema generadas a partir de la base de datos.

En el FrontEnd, se encuentra la estructura Web conformada por HTML, JavaScript y CSS, así como el framework principal utilizando que es AngularJS, donde también se incluyen los Módulos y Controladores necesarios para su comunicación con el BackEnd.

### 2.8.1 Atributos

Los nombres de los atributos llevarán su nombre escrito en 'lowerCamelCase' esto significa escribir todas las palabras juntas ubicando la primera letra en minúscula mientras que la primera letra de las palabras subsiguientes, se escribe en mayúscula. Ejemplo: "especimenPersonal". En lo posible se procurará tener nombres de atributos conformados por una sola palabra para evitar extender el nombre más de lo necesario.

### 2.8.2 Clases

Los nombres de las Clases, al igual que en el modelo de la base de datos, llevarán su nombre en 'CamelCase' tradicional, esto quiere decir, escribiendo todas las palabras junto y la primera letra de cada palabra en mayúscula incluyendo la primera. Ejemplo: "EspecieAreaProtegida".

Las Clases básicas del programa, al ser generadas usando ADO.NET, Tienen el mismo nombre de las tablas de las entidades de la base de datos, sin embargo, toda clase que deba ser creada a futuro, seguirá el estándar descrito anteriormente.

### 2.8.3 Controladores

Cada nombre de controlador, tanto para AngularJS como para la WebAPI en C#, deberá seguir el estándar de las Clases, Adicionalmente, después del nombre del objeto al que va a estar asociado, se debe agregar la palabra 'Controller', es decir, si vamos a crear un controlador para Especie, tendrá el siguiente nombre: "EspeciesController".

Para agrupar diferentes controladores que corresponden a una misma categoría, se los agregará dentro de una carpeta que llevará el nombre del área a la cual pertenecen los controladores. Por ejemplo, dentro de la carpeta: "TaxonomyControllers" se encontrarán controladores como: "EspeciesController", "ReinosController", "FamiliasController", entre otros.

#### 2.8.4 Vistas

Los archivos HTML para las vistas deberán ser nombrados dependiendo el tipo de formulario. Para un nuevo registro, deberá llevar “new” al inicio, “edit” para edición y “ver” para consulta, agregando el nombre del objeto en que se trabajará.

#### 2.8.5 Métodos

Los nombres de los métodos de selección y acción del BackEnd, desarrollados en C#, serán escritos en ‘lowerCamelCase’. Los métodos de inserción tendrán el nombre del verbo HTTP “más semejante” añadiendo el nombre del objeto que se piense insertar, es decir se escribirá como: “Post<objeto>”.

Siguiendo el mismo modelo, las consultas serán escritas como: “Get<objeto>”, las actualizaciones, se escribirán como: “Put<objeto>” y, finalmente, las eliminaciones serán escritas como “Delete<objeto>”.

En el caso de las actualizaciones, consultas y eliminaciones que dependan de otras clases, los nombres de los métodos harán referencias a dichas clases anteponiendo la palabra ‘from’ seguida del nombre de la clase u objeto. Ejemplo: “DeleteAudioFromEspecie”, “GetAudioFromEspecie”, “PutAudioFromEspecie”.

Los nombres de los métodos de selección y acción del FrontEnd, desarrollados con AngularJS, serán nombrados utilizando los verbos “get”, “add”, “save” y “delete”, añadiendo el nombre del objeto que se piense insertar, es decir se escribirá como: “addEspecimen”, “getEspecimen”, “saveEspecimen”, “deleteAutor”.

Estos métodos serán llamados desde botones ubicados en cada una de las vistas de la Web.

En el caso de los Scripts de SQL para la base de datos, los stored procedures o procedimientos almacenados, llevaran “SP\_” antes del nombre del mismo.

#### 2.8.6 Organización de archivos y carpetas para Página Web

La organización de archivos y carpetas en el FrontEnd, se realizará de la siguiente manera:

Los archivos y carpetas que están relacionados con el proyecto de la página web, tales como archivos JavaScript, CSS, imágenes, fuentes de estilos de textos y archivos de AngularJS, se ubicarán dentro de la carpeta llamada “dist”.

Dentro de dicha carpeta, se ubicarán las carpetas, “css”, “font”, “js”, “img”, donde se ubicarán los archivos que correspondan a dichas categorías.

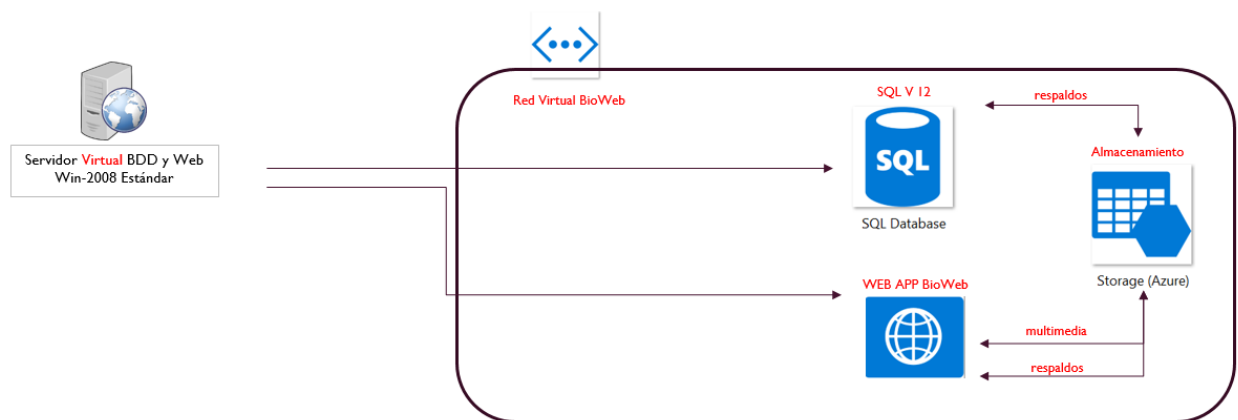
En la carpeta “js”, también se ubicarán los archivos de AngularJS incluyendo los archivos base, controladores y servicios.

En la raíz del directorio, en la carpeta “app” y dentro la carpeta “views”, se ubicará a los archivos HTML con la implementación de las interfaces del sistema Web.

## Capítulo 3: Desarrollo del Sistema

### 3.1 Arquitectura

En la imagen mostrada a continuación, se detalla la arquitectura que se usará en el proyecto.



*Ilustración 5 Arquitectura del Proyecto*

El Servidor de base de datos y web actual, se reemplazará por servicios alojados en Azure, lo que garantizará su alta disponibilidad y escalabilidad.

### 3.2 Plataforma

El esquema de las plataformas a ser usado en el desarrollo del proyecto es el siguiente:

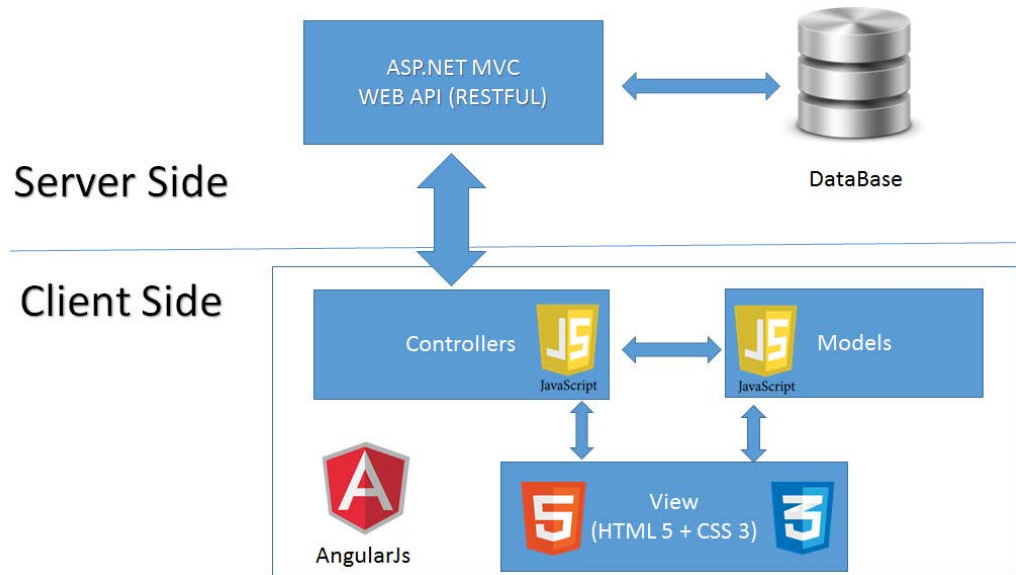


Ilustración 6 Tecnología Utilizada (Omar N., 2017)

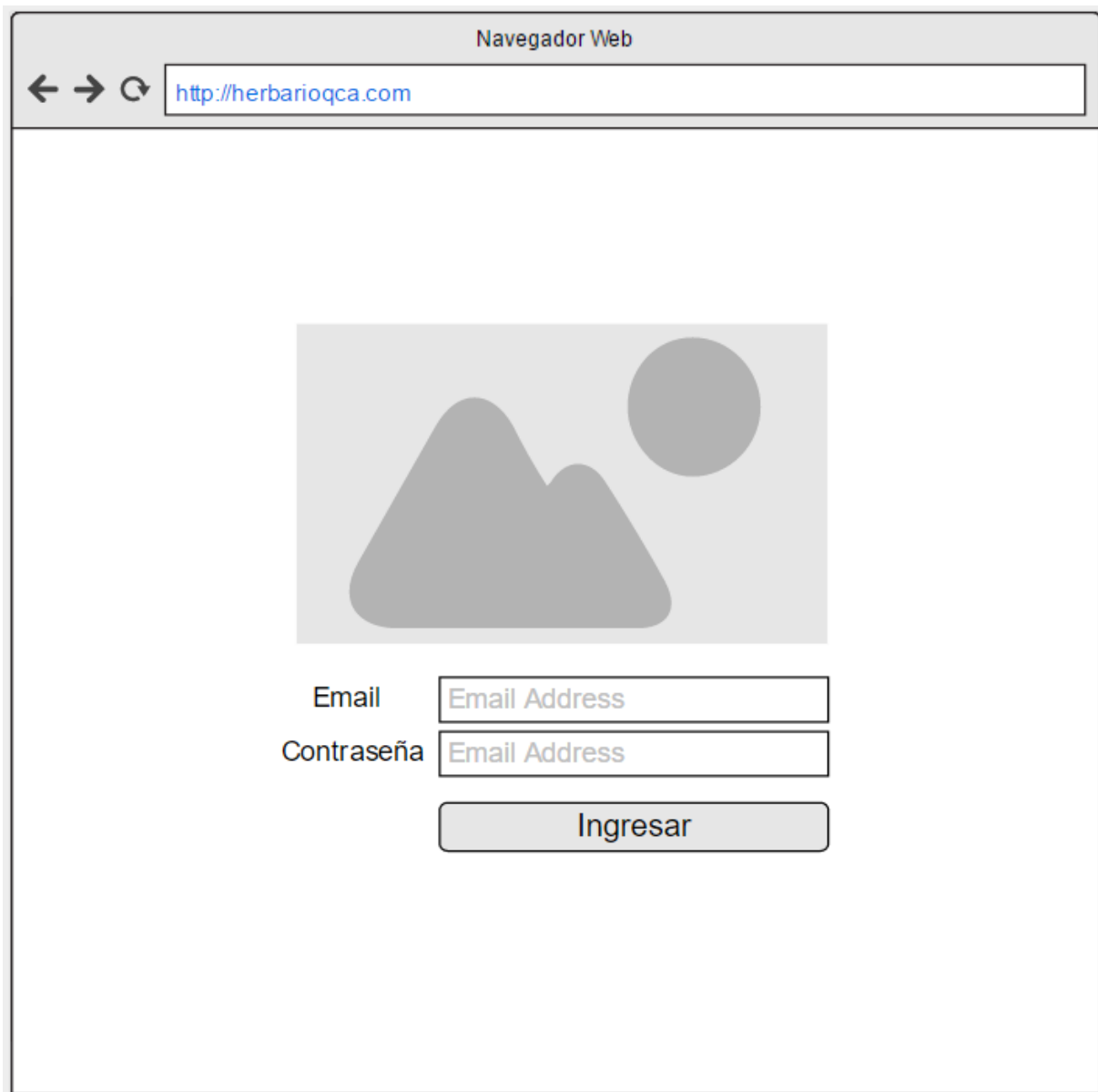
La tecnología utilizada, denominada Simple Page Application o Aplicación de una Sola Página implica ciertos retos que no están presentes en las aplicaciones web tradicionales. Sin embargo, las tecnologías emergentes como ASP.NET Web API, frameworks de JavaScript como AngularJS y nuevas funciones de estilo proporcionadas por CSS3 hacen que sea realmente fácil de diseñar y construir una SPA.

El proyecto aprovechará esas tecnologías para implementar la capa de servicio con ASP.NET Web API el cual será consumido por la aplicación desarrollada utilizando una interfaz de usuario agradable y sensible utilizando los efectos de transformación AngularJS y CSS3.

### 3.3 Diseño de interfaces

Para realizar el diseño preliminar de las interfaces, se utilizó una herramienta online llamada “Moqups” (“Online Mockup, Wireframe & UI Prototyping Tool - Moqups”, 2017, Alojada en: <https://moqups.com>) para realizar los bocetos de las interfaces básicas del sistema Web.



A continuación, se exhibe el diseño de las interfaces antes de ser programadas:



*Ilustración 7 Interface Inicio de Sesión*

Navegador Web

← → ↻ <http://herbarioqca.com>

 Usuario 

Menu1 ▲

- All
- Submenu1
- Submenu2
- Submenu3

Home > Nivel 1 > Nivel 2 > Nivel 3 > Actual

Acordeon 1 ▼

Acordeon 2 ▼

Dato1

Texto

Check1  Si


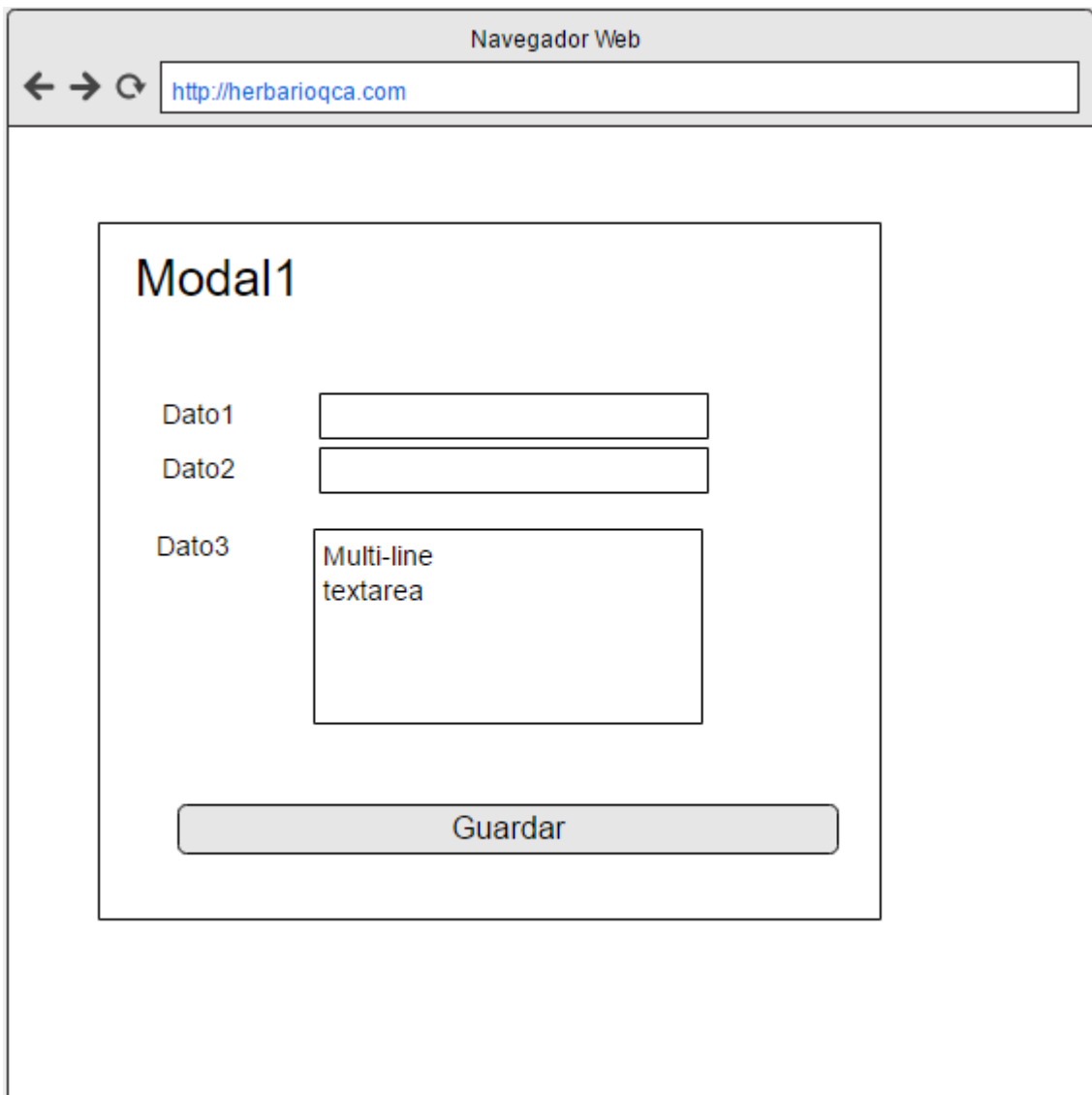
Fecha  

Ilustración 8 Interface Formulario



*Ilustración 9 Interface Modal*

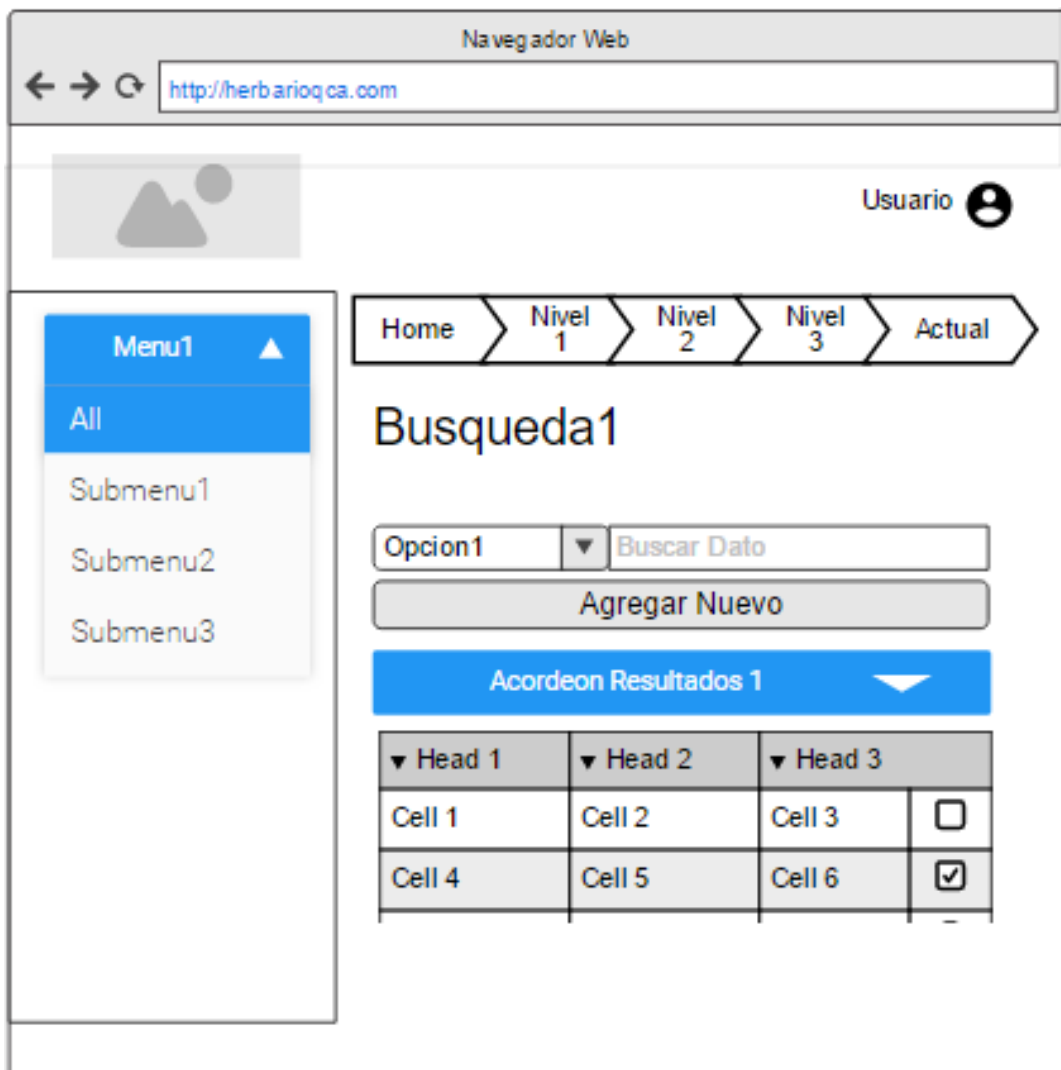


Ilustración 10 Interface Búsqueda y Resultado

### 3.4 Diagramas de paquetes

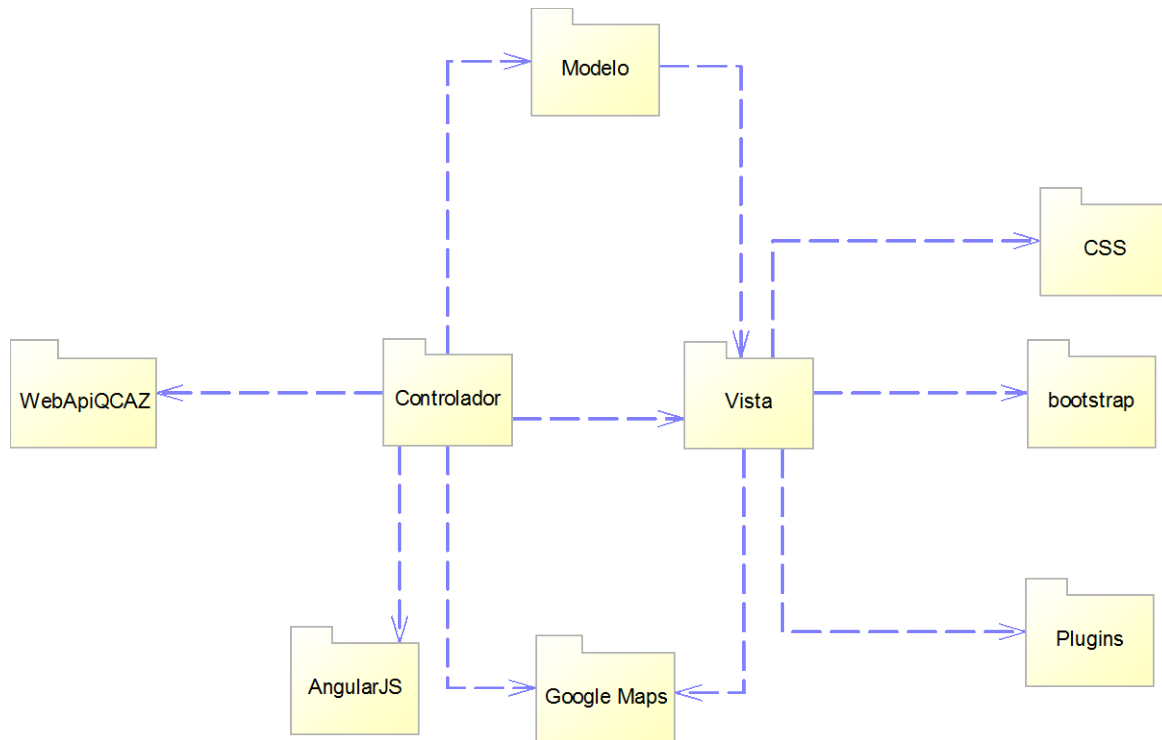


Diagrama 10 Paquetes AngularJS WebApiQCAZ

### 3.5 Diagramas de despliegue

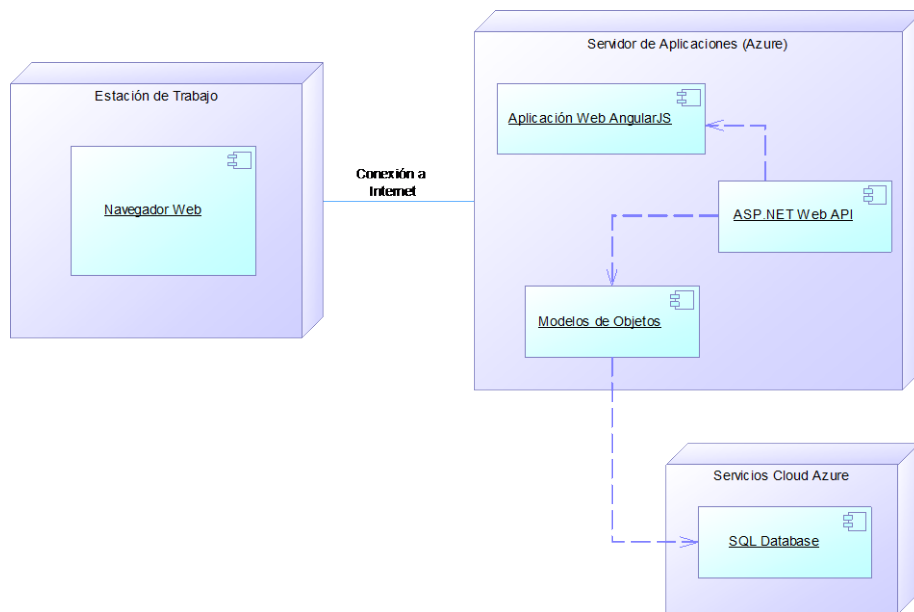


Diagrama 11 Despliegue del Sistema

### 3.6 Sprint 1

#### 3.6.1 Descripción

Para el primer Sprint, se fijó un tiempo de 4 semanas, en este tiempo, se realizará la implementación de las historias de usuario 1 y 3, las cuales tienen que ver con la gestión de accesos al Sistema y la gestión de contenidos del sitio respectivamente.

También, en este Sprint se realizará el modelado físico de la base de datos del Sistema.

#### 3.6.2 Pila del producto Sprint 1

En la tabla ubicada a continuación, se especifica la pila del producto que se definió para el desarrollo del Sprint 1.

Pila del Producto para Sprint 1			
<b>Número</b>	1	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Autenticación de Usuarios.		
<b>Prioridad en Negocio</b>	Alta		
<b>Descripción</b>	Se requiere que el sistema permita ingresar solo a usuarios registrados en el mismo mediante un nombre y una contraseña, y pueda acceder a diferentes opciones de acuerdo a un rol.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"><li>• Ingresar al sistema con su nombre usuario.</li><li>• El sistema muestra las opciones de menú y campos de formularios de acuerdo al rol.</li></ul>		
<b>Número</b>	3	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Contenidos.		
<b>Prioridad en Negocio</b>	Media.		
<b>Descripción</b>	Mostrar información general del Herbario, similar a lo ya expuesto en puce.edu.ec. Permitir la edición del contenido de la página.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"><li>• Presentar la información del herbario en secciones, similar a la página puce.edu.ec.</li></ul>		

*Tabla 14 Pila del Producto Sprint 1  
Autores: Daniel Angulo & Diego Freire*

### 3.6.3 Actividades del Sprint 1:

En esta fase, se detallan las actividades o tareas a realizar en la iteración 1, las cuales englobarán a las historias de usuario 1 (Gestión de Accesos), 3 (Gestión de contenidos) y el Modelado de Base de datos, estas actividades son responsabilidad del equipo de desarrollo.

<b>Sprint: 1</b>		<b>Fecha Inicial:</b> 19/09/2016	<b>Duración:</b> 20 días		<b>Días</b>	19/09/2016	26/09/2016	3/10/2016	10/10/2016
<b>ID Historia Usuario</b>	<b>Tarea</b>	<b>Responsable</b>	<b>Estado</b>	<b>Observación</b>	<b>Horas</b>	<b>20</b>	<b>20</b>	<b>20</b>	<b>20</b>
1	Modelado y diseño de Base de Datos inicial.	Equipo	Completado			X	X	X	X
1	Elaboración de diseño de pantallas de navegación.	Equipo	Completado					X	
1	Elaboración de Formularios de ingreso.	Equipo	Completado					X	
1	Codificación de página de Acceso de Usuarios.	Equipo	Completado					X	
1	Codificación de controladores para gestión de Usuarios.	Equipo	Completado					X	
1	Pruebas de funcionamiento para implementación de página de acceso y control de usuarios.	Equipo	Completado						X
3	Elaboración de diseño de pantallas y formulario para creación de contenido.	Equipo	Completado						X

3	Elaboración de secciones y categorías para el contenido.	Equipo	Completado							X
3	Codificación de página de gestión de contenido.	Equipo	Completado							X
3	Codificación de controladores para gestión de contenido.	Equipo	Completado							X
3	Pruebas de funcionamiento para implementación de página de gestión de contenidos.	Equipo	Completado							X

*Tabla 15 Pila de actividades Sprint 1*

*Autores: Daniel Angulo & Diego Freire*

### 3.6.4 Pruebas de la Sprint 1

En las siguientes tablas, se detallan las pruebas realizadas para verificar las funcionalidades de las historias de usuarios elaboradas a lo largo de la iteración 1.

**Nota:** Se exhiben las funcionalidades detalladas en los casos de uso para lograr mejor comprensión con el usuario.

**Historia de Usuario 1:** Gestión de Accesos al Sistema.

**Fecha:** 12/10/2016

Caso de uso	Prueba	Resultado Esperado	Estado	Observación
<b>F1.1</b> <b>Iniciar Sesión</b>	Usuario con credenciales intenta acceder a la administración del sistema.	El sistema debe mostrar la administración del sistema habilitando las secciones específicas del rol del usuario.	Correcto	En caso de credenciales incorrectas, se mostrará un mensaje de alerta.
<b>F1.2</b> <b>Registrar Usuarios</b>	El Administrador del sistema agrega un nuevo usuario y rol.	El nuevo usuario, puede iniciar sesión en el sistema y realizar tareas específicas de su rol.	Correcto	Todo usuario es registrado con su Correo Electrónico.

<b>F1.3.1</b> <b>Modificar Cuenta Usuario</b>	El Administrador del sistema edita datos de un usuario existente.	Los datos editados se ven reflejados en la cuenta del usuario.	Correcto	Un usuario puede ser editado las veces que sea necesario.
<b>F1.3.2</b> <b>Modificar Roles Usuario</b>	El Administrador del sistema edita el rol de un usuario existente.	El rol editado se ve reflejado en la cuenta del usuario y en su nivel de acceso.	Correcto	Un usuario puede ser editado las veces que sea necesario.
<b>F1.5.1</b> <b>Consultar Todos</b>	El Administrador del sistema consulta todos los usuarios registrados.	El sistema muestra los datos de los usuarios registrados.	Correcto	Los resultados pueden ser filtrados por área, rol o demás datos.

*Tabla 16 Pruebas Gestión de Accesos al Sistema*

*Autores: Daniel Angulo & Diego Freire*

### Historia de Usuario 3: Gestión de Contenido.

Fecha: 12/10/2016

<b>Caso de uso</b>	<b>Prueba</b>	<b>Resultado Esperado</b>	<b>Estado</b>	<b>Observación</b>
<b>F7.1</b> <b>Creación de Paginas</b>	Guardar Pagina.	Crear una página en base a una plantilla con su título y una imagen.	Correcto	Ninguna
<b>F7.3</b> <b>Ingreso de Contenido</b>	Seleccionar una página y agregar contenido.	Crear un post y agregar su contenido.	Correcto	Ninguna
<b>F7.4</b> <b>Modificación de Contenido</b>	Seleccionar una página, seleccionar un post y modificar su contenido.	Modificar los datos del post.	Correcto	Ninguna

*Tabla 17 Pruebas Gestión de Contenido del Sitio*

*Autores: Daniel Angulo & Diego Freire*

### 3.7 Sprint 2

#### 3.7.1 Descripción

Tras haberse realizado el primer Sprint, se fijó un tiempo de 4 semanas para el Sprint 2, en este tiempo, se abordará la elaboración de las historias de usuario 2 y 4, las cuales corresponden con la gestión de taxones y la gestión de localidades respectivamente.

También, en este Sprint se realizará una depuración del modelo físico de la base de datos del Sistema para aumentar campos necesarios para la información de localidades.

#### 3.7.2 Pila del producto Sprint 2

En la tabla ubicada a continuación, se especifica la pila del producto que se definió para el desarrollo del Sprint 2.

Pila del Producto para Sprint 2			
<b>Número</b>	2	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Taxonomía.		
<b>Prioridad en Negocio</b>	Alta		
<b>Descripción</b>	Permitir consultar la taxonomía de una especie. Además, se podrán ingresar nuevos registros y actualizar los ya ingresados.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"><li>• Ingresar una especie en la base de datos, seleccionando su taxonomía de manera jerárquica.</li><li>• En caso de no tener la taxonomía completa, se ingresa los taxones faltantes.</li><li>• Actualizar los datos de una especie.</li><li>• Consultar la taxonomía de una especie.</li><li>• Consultar la ficha de una especie.</li></ul>		
<b>Número</b>	4	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Localidades.		
<b>Prioridad en Negocio</b>	Alta		
<b>Descripción</b>	Se requiere que el sistema permita el ingreso, consulta y actualización de localidades por medio un formulario que despliegue las divisiones geográficas jerárquicamente.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"><li>• Ingresar una localidad en la base de datos.</li><li>• En caso de no existir información completa de las divisiones geográficas superiores, se ingresa la división geográfica correspondiente.</li></ul>		

- Actualizar los datos de una localidad.
- Consultar localidades.
- Consultar detalles de una localidad.

Tabla 18 Pila del Producto para Sprint 2

Autores: Daniel Angulo & Diego Freire

### 3.7.3 Actividades del Sprint 2:

En esta fase, se detallan las actividades o tareas a realizar en la iteración 2, las cuales englobarán a las historias de usuario 2 (Gestión de Taxones), 4 (Gestión de Localidades) y la depuración del modelo de Base de datos, estas actividades son responsabilidad del equipo de desarrollo.

Sprint: 2		Fecha Inicial: 17/10/2016	Duración: 20 días		Días	17/10/2016	24/10/2016	31/10/2016	7/11/2016
ID Historia Usuario	Tarea	Delegado	Estado	Observación	Horas	20	20	20	20
2,4	Depuración Base de Datos.			La base de datos se encuentra en constante evolución debido a las necesidades de los usuarios.		X	X	X	X
2	Elaboración de diseño de pantallas de navegación.	Equipo	Completado			X			
2	Elaboración de Formularios de ingreso de taxones superiores.	Equipo	Completado			X			
2	Elaboración de diseño de Formularios de	Equipo	Completado			X			

	ficha de especie.							
2	Codificación de ficha de especie y taxonomía superior.	Equipo	Completado		X			
2	Codificación de controladores para gestión de Taxonomía.	Equipo	Completado			X		
2	Pruebas de funcionamiento para implementación de Gestión de Taxones.	Equipo	Completado			X		
4	Elaboración de diseño de pantallas de navegación.	Equipo	Completado				X	
4	Elaboración de Formularios individuales de divisiones geográficas.	Equipo	Completado				X	
4	Elaboración de diseño de Formularios de localidad.	Equipo	Completado				X	

4	Codificación de página de gestión de localidades.	Equipo	Completado				X	
4	Codificación de controladores para gestión de Localidades.	Equipo	Completado				X	X
4	Pruebas de funcionamiento para implementación de página de gestión de localidades.	Equipo	Completado					X

Tabla 19 Pila de actividades Sprint 2

Autores: Daniel Angulo & Diego Freire

### 3.7.4 Pruebas de la Sprint 2

En las siguientes tablas, se detallan las pruebas realizadas para verificar las funcionalidades de las historias de usuarios elaboradas a lo largo del Sprint 2.

**Nota:** Se exhiben las funcionalidades detalladas en los casos de uso para lograr mejor comprensión con el usuario.

**Historia de Usuario 2:** Gestión de Taxones.

**Fecha:** 9/11/2016

Caso de uso	Prueba	Resultado Esperado	Estado	Observación
<b>F2.1</b> <b>Ingresar Taxones</b>	Registrar una nueva taxonomía empezando desde Reino.	El taxón ingresado debe ser visible en la lista de su categoría.	Correcto	Para ingresar una categoría taxonómica inferior, primero debe ingresarse el taxón padre.
<b>F2.2.1</b> <b>Modificar Taxones</b>	Modificar un taxón ya existente en el sistema.	Los datos del taxón editado deben ser reflejados en el sistema.	Correcto	Ninguna

<b>F2.2.2</b> <b>Modificar Ficha Especie</b>	Modificar una ficha de especie ya existente en el sistema.	Los datos de la ficha de especie editada, deben ser reflejados en el sistema.	Correcto	La taxonomía completa de dicha ficha, también puede ser editada en este menú.
<b>F2.4</b> <b>Consultar Taxones</b>	Consultar la taxonomía de una especie ingresada al sistema.	Al Consultar la taxonomía desde el Reino, las listas siguientes deben desplegar la categoría más particular a medida que se vaya avanzando.	Correcto	Ninguna.

*Tabla 20 Pruebas Gestión de Taxones*

*Autores: Daniel Angulo & Diego Freire*

### Historia de Usuario 3: Gestión de Localidades

Fecha: 9/11/2016

Caso de uso	Prueba	Resultado Esperado	Estado	Observación
<b>F3.1</b> <b>Ingresar Localidades</b>	Seleccionar continente, país, provincia, cantón, parroquia para definir una localidad.	Seleccionar un continente y obtener sus países, seleccionar un país y obtener sus provincias, seleccionar una provincia para obtener sus islas y cantones, seleccionar un cantón para obtener las parroquias.	Correcto	
	Registrar una localidad.	Ingresar datos de la localidad, confirmar su ubicación en el mapa y guardar.	Correcto	
<b>F3.2.1</b> <b>Modificar Divisiones Geográficas</b>	Seleccionar cada región como continente, país, provincia, parroquia, isla para su modificación.	Seleccionar cada región y obtener datos de forma jerárquica, y modificar cada uno de ellos.	Correcto	

<b>F3.2.2</b> <b>Modificar Detalle</b> <b>Localidad</b>	Seleccionar una localidad y modificar sus datos.	Obtener los datos de la localidad seleccionada, modificarlos y guardar los cambios.	Correcto	
<b>F3.4</b> <b>Consultar Localidad</b>	Consultar localidades de acuerdo a un filtro por regiones geográficas.	Obtener una lista de localidades por continente, país, provincia, parroquia y nombre de localidad	Correcto	
	Seleccionar una localidad para ver información detallada.	Al seleccionar la localidad se debe abrir la página verLocalidad con los datos de la misma para su visualización.	Correcto	

*Tabla 21 Pruebas Gestión de Localidades*

*Autores: Daniel Angulo & Diego Freire*

## 3.8 Sprint 3

### 3.8.1 Descripción

El Sprint 3 tendrá una duración de 4 semanas, en este tiempo, se continuará la elaboración de las historias de usuario 2, pero se empezará con la 6. Estas historias de usuario corresponden a gestión de taxones y gestión de publicaciones respectivamente.

También, en este Sprint se realizará una depuración del modelo físico de la base de datos del Sistema para aumentar campos necesarios para la información de localidades.

### 3.8.2 Pila del Producto Sprint 3

En la tabla ubicada a continuación, se especifica la pila del producto que se definió para el desarrollo del Sprint 3.

<b>Pila del Producto para Sprint 3</b>			
<b>Número</b>	2	<b>Usuario</b>	Usuario Registrado
<b>Nombre de la Historia</b>	Gestión de Taxonomía		
<b>Prioridad en Negocio</b>	Alta		
<b>Descripción</b>	Permitir consultar la taxonomía de una especie. Además, se podrán ingresar nuevos registros y actualizar los ya ingresados.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Ingresar una especie en la base de datos, seleccionando su taxonomía de manera jerárquica.</li> <li>• En caso de no tener la taxonomía completa, se ingresa los taxones faltantes.</li> <li>• Actualizar los datos de una especie.</li> <li>• Consultar la taxonomía de una especie.</li> <li>• Consultar la ficha de una especie.</li> </ul>		
<b>Número</b>	6	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Publicaciones.		
<b>Prioridad en Negocio</b>	Alta		
<b>Descripción</b>	Se requiere que el sistema permita el ingreso, consulta y actualización de publicaciones científicas por medio un formulario, y la información a de su autor.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"> <li>• Ingresar una publicación y seleccionar autor.</li> <li>• Actualizar los datos de una publicación.</li> <li>• Ingresar autores.</li> <li>• Actualizar autores.</li> </ul>		

*Tabla 22 Pila del Producto para Sprint 3  
Autores: Daniel Angulo & Diego Freire*

### 3.8.3 Actividades de Sprint 3

En esta fase, se detallan las actividades o tareas a realizar en el Sprint 3, las cuales englobarán a las historias de usuario 6 (Gestión de Publicaciones), 2 (Gestión de Taxonomía) y la depuración del modelo de Base de datos, estas actividades son responsabilidad del equipo de desarrollo.

<b>Sprint: 3</b>		<b>Fecha Inicial:</b> 14/11/2016	<b>Duración:</b> 20 días		<b>Días</b>	14/11/2016	21/11/2016	28/11/2016	5/12/2016
<b>ID Historia de Usuario</b>	<b>Tarea</b>	<b>Delegado</b>	<b>Estado</b>	<b>Observación</b>	<b>Horas</b>	20	20	20	20
2,6	Depuración Base de Datos.	Equipo	Completado	La base de datos se encuentra en constante evolución debido a las necesidades de los usuarios.		X	X	X	X
6	Generación de Recurso en API.	Equipo	Completado	Es necesario reflejar los cambios hechos en la base de datos		X	X	X	X
6	Codificación de Formulario para ingreso de Publicaciones.	Equipo	Completado			X			
6	Codificación de Controlador de Publicaciones.	Equipo	Completado			X			
6	Codificación de Formulario de Edición de Publicaciones.	Equipo	Completado			X			

6	Codificación de Pagina de Búsqueda de Publicaciones.	Equipo	Completado			X		
6	Pruebas de funcionamiento de formularios.	Equipo	Completado			X		
2	Codificación de Formulario para ingreso de Autores Taxa.	Equipo	Completado			X		
2	Codificación de Formulario para edición de Autor Taxa.	Equipo	Completado				X	
2	Codificación de Pagina de Búsqueda de Autor Taxa.	Equipo	Completado				X	
2	Modificación de Controlador Taxonomía.	Equipo	Completado					X
2	Pruebas de funcionamiento para Autor Taxa.	Equipo	Completado					X

*Tabla 23 Pila de Actividades del Producto Sprint 3  
Autores: Daniel Angulo & Diego Freire*

### 3.8.4 Pruebas de Sprint 3

En las siguientes tablas, se detallan las pruebas realizadas para verificar las funcionalidades de las historias de usuarios elaboradas a lo largo del Sprint 3.

#### Historia de Usuario 6: Gestión de Publicaciones Científicas.

Fecha: 7/12/2016

Caso de uso	Prueba	Resultado Esperado	Estado	Observación
<b>F5.1</b> <b>Ingresar Publicaciones Científicas</b>	Cargar los datos de las listas.	Al abrir la página NuevaPublicación, las listas para selección deben cargar datos.	Correcto	En caso de agregar nuevos datos para las listas, se debe presionar el botón refrescar para actualizar la lista.
	Guardar Publicación.	Insertar los datos y archivos de la publicación en la base de datos.	Correcto	Ninguna
<b>F5.2</b> <b>Modificar Publicaciones Científicas</b>	Abrir la página de edición de la publicación con sus respectivos datos.	Al elegir editar una de las publicaciones de una lista de consulta, se abre la página EditarPublicación con los datos correspondientes a la publicación	Correcto	Ninguna
	Actualizar Publicación	Realizar actualización de la publicación en la base de datos.	Correcto	Ninguna
<b>F5.4</b> <b>Consultar Publicaciones Científicas</b>	Consulta de publicaciones por título, autor, editorial o año de publicación.	Obtener una lista de publicaciones de acuerdo al campo de búsqueda	Correcto	Se puede hacer un filtrado adicional de las listas obtenida en la consulta.
	Seleccionar una publicación y ver más detalles.	Observar los datos completos de la publicación en la página VerPublicación.	Correcto	

*Tabla 24 Pruebas de Gestión de Publicaciones  
Autores: Daniel Angulo & Diego Freire*

## Historia de Usuario 2: Gestión de Taxones

Fecha: 7/12/2016

Caso de uso	Prueba	Resultado Esperado	Estado	Observación
<b>F2.1</b> <b>Ingresar Taxones</b>	Registrar una nueva taxonomía agregando el autor.	El taxón ingresado debe ser visible en la lista de su categoría.	Correcto	Para ingresar una categoría taxonómica inferior, primero debe ingresarse el taxón padre.
<b>F2.2.1</b> <b>Modificar Taxones</b>	Modificar un taxón ya existente en el sistema y agregar su autor.	Los datos del taxón editado deben ser reflejados en el sistema.	Correcto	Ninguna
<b>F2.2.2</b> <b>Modificar Ficha Especie</b>	Modificar una ficha de especie ya existente en el sistema, incluyendo el autor.	Los datos de la ficha de especie editada, deben ser reflejados en el sistema.	Correcto	La taxonomía completa de dicha ficha, también puede ser editada en este menú.
<b>F2.4</b> <b>Consultar Taxones</b>	Consultar la taxonomía de una especie ingresada al sistema verificando el autor ingresado.	Al Consultar la taxonomía desde el Reino, las listas siguientes deben desplegar la categoría más particular a medida que se vaya avanzando.	Correcto	Ninguna.

Tabla 25 Pruebas de Gestión de Taxones

### 3.9 Sprint 4

#### 3.9.1 Descripción

Para el desarrollo de este sprint, se utilizarán las especificaciones de la pila del producto definidas al inicio del proyecto. Este Sprint tendrá una duración de 4 semanas, al igual que los anteriores.

#### 3.9.2 Pila del Producto Sprint 4

En la tabla ubicada a continuación, se especifica la pila del producto que se definió para el desarrollo del Sprint 4.

<b>Pila del Producto Sprint 4</b>			
<b>Número</b>	7	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Préstamos.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Se requiere que el sistema permita el registro, consulta y actualización de préstamos realizados por el personal del herbario por medio un formulario, y la información del beneficiario.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"><li>• Registrar un préstamo y seleccionar el beneficiario.</li><li>• Actualizar el estado del préstamo.</li><li>• Ingresar beneficiarios.</li><li>• Actualizar beneficiarios.</li></ul>		
<b>Número</b>	5	<b>Usuario</b>	Usuario Registrado.
<b>Nombre de la Historia</b>	Gestión de Espécimen.		
<b>Prioridad en Negocio</b>	Alta.		
<b>Descripción</b>	Se requiere que el sistema permita el ingreso, consulta y actualización de especímenes por medio un formulario que despliegue la información necesaria para la identificación de la muestra.		
<b>Como Probarlo</b>	<ul style="list-style-type: none"><li>• Ingresar un espécimen asociado a una especie.</li><li>• Actualizar los datos de un espécimen.</li><li>• Consultar especímenes.</li></ul>		

*Tabla 26 Pila del Producto Sprint 4  
Autores: Daniel Angulo & Diego Freire*

### 3.9.3 Actividades del Sprint 4

En esta fase, se detallan las actividades o tareas a realizar en el Sprint 4, las cuales englobarán a las historias de usuario 5 y 7, correspondientes a Gestión de Especímenes y Gestión de Préstamos, además de la depuración en la base de datos.

Sprint: 4		Fecha Inicial: 12/12/2016	Duración: 20 días		Días	12/12/2016	02/01/2017	09/01/2017	16/01/2017
ID Historia de Usuario	Tarea	Delegado	Estado	Observación	Horas	20	20	20	20
7,5	Depuración Base de Datos.	Equipo	Completado	La base de datos se encuentra en constante evolución debido a las necesidades de los usuarios		X	X	X	X
7	Generación de Recurso en API.	Equipo	Completado			X	X	X	X
7	Codificación de Formulario para ingreso de Préstamos y Beneficiarios.	Equipo	Completado			X			
7	Codificación de Controlador de Préstamos.	Equipo	Completado			X			
7	Codificación de Formulario de Edición de Préstamos y Beneficiarios.	Equipo	Completado				X	X	

7	Codificación de Pagina de Búsqueda de Préstamos y Beneficiarios.	Equipo	Completado				X	
7	Pruebas de funcionamiento de formularios.	Equipo	Completado				X	
5	Modificación de Controlador Especímenes.	Equipo	Completado				X	
5	Codificación de Formulario para edición de Especímenes.	Equipo	Completado				X	
5	Codificación de Pagina de Búsqueda de Especímenes.	Equipo	Completado					X
5	Codificación de Formulario para ingreso de Especímenes.	Equipo	Completado					X
5	Pruebas de funcionamiento para Especímenes.	Equipo	Completado					X

*Tabla 27 Actividades de Sprint 4  
Autores: Daniel Angulo & Diego Freire*

### 3.9.4 Pruebas de Sprint 4

En las siguientes tablas, se detallan las pruebas realizadas para verificar las funcionalidades de las historias de usuarios elaboradas a lo largo del Sprint 4.

#### Historia de Usuario 7: Gestión de Prestamos

Fecha: 18/01/2017

Caso de Uso	Prueba	Resultado Esperado	Estado	Observación
<b>F6.1</b> <b>Ingresar Préstamo</b>	Seleccionar responsable del préstamo, beneficiario, país de origen, país destino y país retorno. Ingresar datos del préstamo.	Seleccionar los datos requeridos de las listas e ingresar los datos del préstamo.	Correcto	
	Añadir Préstamo	Guardar los datos básicos del préstamo.	Correcto	
<b>F6.2</b> <b>Modificar Préstamo</b>	Seleccionar un préstamo en la búsqueda para su edición y obtener los datos del mismo	Seleccionar los datos requeridos de las listas e ingresar los datos del préstamo.	Correcto	
	Actualizar Préstamo	Guarda los datos modificados en el formulario.	Correcto	
<b>F6.3</b> <b>Consultar Préstamo</b>	Buscar un préstamo por Espécimen o código QCAZ	Obtener una lista de préstamos de acuerdo al tipo de búsqueda.	Correcto	

*Tabla 28 Pruebas de Gestión de Préstamos  
Autores: Daniel Angulo & Diego Freire*

## Historia de Usuario 5: Gestión de Especímenes

Fecha: 18/01/2017

Caso de uso	Prueba	Resultado Esperado	Estado	Observación
<b>F4.1</b> <b>Ingresar Especimen</b>	Cargar los datos de las listas.	Al iniciar la página se deben cargarse las listas para la selección.	Correcto	En caso de agregar nuevos datos para las listas, se debe presionar el botón refrescar para actualizar la lista.
	Registrar un nuevo espécimen.	Ingresar datos del Especimen a detalle, subir imágenes, seleccionar una localidad.	Correcto	Ninguna
<b>F4.2</b> <b>Modificar Especimen</b>	Modificar los datos de un espécimen.	Cargar los datos del espécimen, realizar y registrar las modificaciones.	Correcto	Ninguna
<b>F4.4</b> <b>Consultar Especimen</b>	Consultar un espécimen por especie.	Cargar una lista con los especímenes de acuerdo a una especie	Correcto	Ninguna
	Seleccionar un espécimen para ver más detalles.	Abrir la página ver Especimen para ver información detallada sobre el espécimen.	Correcto	Ninguna

Tabla 29 Pruebas de Gestión de Especímenes  
Autores: Daniel Angulo & Diego Freire

### 3.10 Sprint 5

#### 3.10.1 Descripción

La pila del producto inicial fue completada en el Sprint 4. Las funcionalidades principales del sistema fueron implementadas correctamente y pasaron las pruebas requeridas. En este sprint se empiezan a desarrollar características adicionales, además se realizarán correcciones solicitadas por los usuarios.

#### 3.10.2 Pila del Producto Sprint 5

En la tabla ubicada a continuación, se especifica la pila del producto que se definió para el desarrollo del Sprint 5.

Pila del Producto				
ID	Nombre de la Historia	Descripción	Sprint	Como Probarlo
5	Gestión de Especímenes	Agregar generador de Código de Barras.	5	<ul style="list-style-type: none"><li>En la página de consulta de especímenes, pulsar un botón que genere un código de barras con el código QCA.</li></ul>

Tabla 30 Pila del Producto Sprint 5  
Autores: Daniel Angulo & Diego Freire

### 3.10.3 Actividades de Sprint 5

Sprint: 5		Fecha Inicial: 23/01/2017	Duración: 20 días		Días	23/01/2016	30/01/2017	6/02/2017	13/07/2017
ID Historia de Usuario	Tarea	Delegado	Estado	Observación	Horas	20	20	20	20
	Depuración Base de Datos.	Equipo	Completado	La base de datos se encuentra en constante evolución debido a las necesidades de los usuarios		X	X	X	X
2,6	Depuración Base de Datos.	Equipo	Completado	La base de datos se encuentra en constante evolución debido a las necesidades de los usuarios		X			
5	Crear Generador de Código de Barras para Especímenes.	Equipo	Completado			X			
5	Agregar Generador de Código de Barras a Búsqueda de Especímenes.	Equipo	Completado			X			

Tabla 31 Actividades de Sprint 5  
Autores: Daniel Angulo & Diego Freire

### 3.10.4 Pruebas de Sprint

En las siguientes tablas, se detallan las pruebas realizadas para verificar las funcionalidades de las historias de usuarios elaboradas a lo largo del Sprint 5.

#### **Historia de Usuario 5: Gestión de Especímenes**

**Fecha:** 1/02/2017

<b>Caso de Uso</b>	<b>Prueba</b>	<b>Resultado Esperado</b>	<b>Estado</b>	<b>Observación</b>
<b>F4.4</b> <b>Consultar Especimen</b>	Consultar un espécimen.	Cargar una lista con los especímenes de acuerdo a los parámetros de búsqueda.	Correcto	Ninguna
	Generar Código de Barras.	Mostrar un código de barras y generar un archivo de imagen para su posterior impresión.	Correcto	Ninguna

*Tabla 32 Pruebas de Consulta Especímenes*

*Autores: Daniel Angulo & Diego Freire*



### 3.12 Diccionario de datos

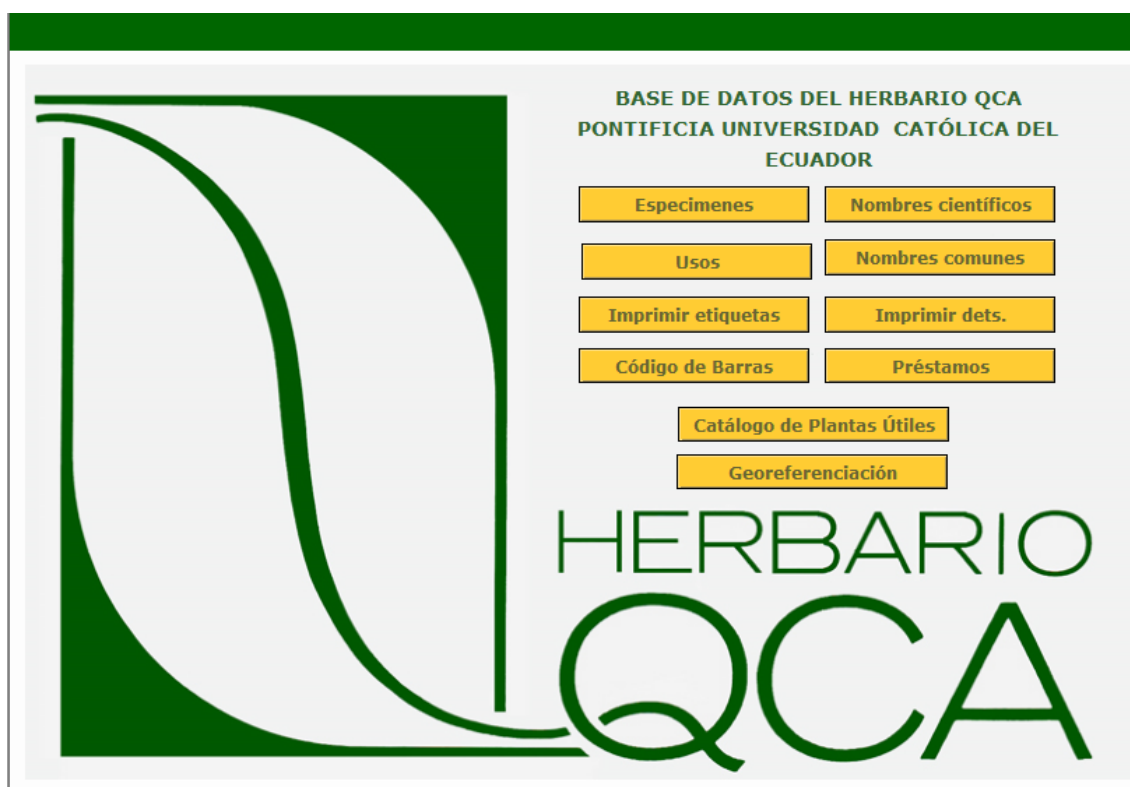
(Anexo 3)

## Capítulo 4: Migración de Base de Datos del HerbarioQCA

### 4.1 Análisis de Base de Datos en FileMaker

Durante el desarrollo de este proyecto, uno de los retos más grandes se presentó al trabajar con la base de datos, desde su diseño hasta la migración.

Para lograr crear un nuevo diseño de base de datos, se analizó el funcionamiento del anterior sistema, que utilizaba FileMaker, el análisis incluyó la determinación de relaciones entre bases de datos internas, tablas que incluye cada una de las bases de datos, campos y registros, además de los formularios para ingreso de la información.



*Ilustración 11 Base de Datos FileMaker Herbario QCA*

Las bases de datos de FileMaker más importantes resultaron ser 'Specimens' y 'SciNames', esto se pudo concluir tras el análisis de las relaciones de tablas y, además, por la información brindada por el personal del HerbarioQCA.

La base de datos 'Specimens', guarda información de los Especímenes, donde se incluyen datos como su localidad de recolección, fecha de recolección, colectores involucrados,

determinación, hábitat, georreferencia, Herbarios donde se encuentra la muestra, partes adicionales, Especímenes tipo, usuario que ingreso la información, fecha de modificación, entre otras cosas.

Gestionar base de datos de "Specimens"

Una tabla es un conjunto único de registros y campos. Un archivo puede contener varias tablas.

11 tablas definidas en este archivo

Nombre de tabla	Origen	Detalles	Apariciones en el gráfico
Specimens	FileMaker	215 campos, 216829 registros	Specimens
Collectors	FileMaker	7 campos, 4010 registros	Collectors
Partes	FileMaker	6 campos, 9118 registros	Partes
GPI_information	FileMaker	5 campos, 1 registro	GPI_information
iabin	FileMaker	198 campos, 354917 registros	iabin
QCA_presentes_Diego	FileMaker	6 campos, 758 registros	QCA_presentes_Diego
Log	FileMaker	14 campos, 63289 registros	Log
Países	FileMaker	3 campos, 250 registros	Países
Incertidumbres	FileMaker	10 campos, 500 registros	Incertidumbres
GeorefLog	FileMaker	16 campos, 110315 registros	GeorefLog
QCA_prestamos	FileMaker	5 campos, 1 registro	QCA_prestamos

Ilustración 12 Base de datos 'Specimens'

Cambiando a la base de datos 'SciNames', se encuentra información sobre la taxonomía jerárquica de cada espécimen, en el formulario de esta base de datos, se ingresa información científica para cada taxón, además de nombres comunes, autores y demás elementos. Cada taxón es relacionado manualmente con la categoría taxonómica superior mediante el ingreso del nombre del taxón padre en un cuadro de texto, lo que daba cabida a que se puedan cometer errores al hacer las relacionar los taxones.

Gestionar base de datos de "SciNames"

Tablas Campos Relaciones

Una tabla es un conjunto único de registros y campos. Un archivo puede contener varias tablas.

5 tablas definidas en este archivo

Nombre de tabla	Origen	Detalles	Apariciones en el gráfico
SciNames	FileMaker	201 campos, 32899 registros	Basionym, Family, homoSyns,
GLOSARIOS	FileMaker	44 campos, 15293 registros	GLOSARIOS
Rank	FileMaker	0 campos, 0 registros	Rank
KeyWords	FileMaker	0 campos, 0 registros	KeyWords
LibroRojo	FileMaker	4 campos, 4442 registros	LibroRojo

Ilustración 13 Base de datos 'SciNames'

Con lo mencionado anteriormente, se pudo determinar que la base de datos 'SciNames' usa un diseño recursivo para asociar la taxonomía, sin embargo, fue muy complicado obtener la jerarquía de cada especie usando esa estructura, debido a que la información ingresada se encuentra, en su gran mayoría, incompleta e inconsistente desde ciertos puntos de la taxonomía.

Una vez concluido este paso, se procedió a diseñar un nuevo modelo de base de datos, que incluya todas las tablas y campos encontrados que tenían mayor importancia, dándoles otro diseño con el fin de mejorar la comprensión a comparación del modelo pasado y también el modo de tratamiento de la información.

El primer modelo de base de datos que se diseñó, estaba basado principalmente en los campos encontrados en los formularios de FileMaker que tenían más información almacenada y también que el personal del HerbarioQCA supo indicar en las visitas que se hizo a la institución. Muchas de las tablas y campos de este diseño inicial fueron agregadas al modelo del Proyecto BIOWEB Ecuador.

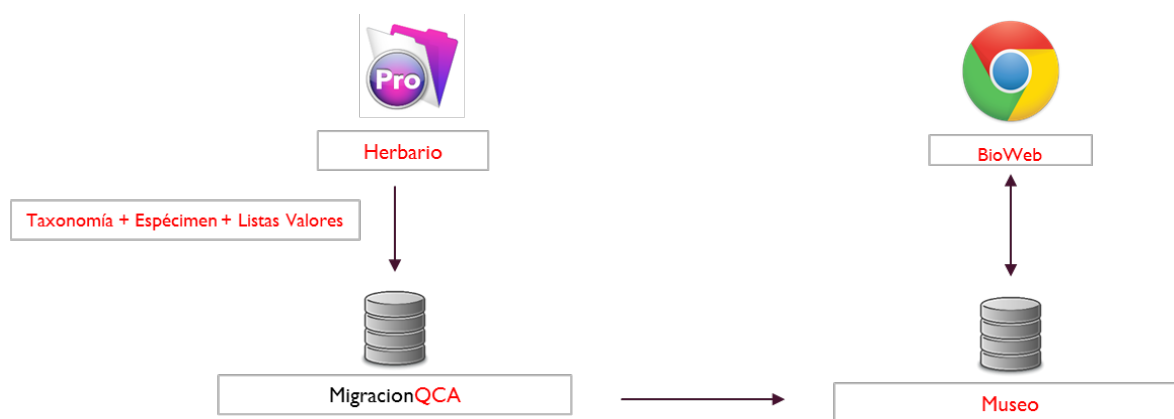
#### 4.2 Diseño de la migración de información

Una vez obtenido el diseño base, se procedió a hacer una planeación para transportar los datos de FileMaker a la nueva base de datos. El proceso de migración fue una de las partes que mayor complejidad tuvo en el desarrollo del proyecto, ya que la información contenida en FileMaker es de gran magnitud (alrededor de 220.000 especímenes) y de suma importancia, por lo que su manipulación se dio con cuidado para evitar pérdida de datos al realizar los scripts de migración.

Para poder exportar los datos desde FileMaker a SQL Server 2014 existen dos alternativas, la primera es compartiendo las tablas de las bases de datos de FileMaker a través del uso de orígenes de datos ODBC, permitiendo su conexión a SQL Server 2014 utilizando LinkedServer dentro de los ServerObjects.

La segunda opción, y más extensa, es exportando los datos de FileMaker a un archivo Excel e importar los datos de este a SQL Server con la ayuda de un asistente, debido a que las bases de datos del Herbario en FileMaker suelen incluir tablas con cientos de campos, se escogió esta opción para bases con tablas específicas que no tienen dimensiones tan grandes como las bases de datos principales.

El esquema de la migración de datos es el siguiente:



*Ilustración 14 Migración de Base de Datos*

Para migrar la información al nuevo diseño, se utilizaron 2 bases de datos en SQL Server 2014. La primera, una base de datos intermedia llamada 'MigracionQCA' donde se almacenan las tablas migradas por LinkedServer o Excel directamente de FileMaker con sus campos convertidos para poder ser usados en SQLServer, además de tablas temporales creadas para facilitar la comprensión e inserción de los datos en las nuevas tablas y campos del Modelo de base de datos del sistema BIOWEB Ecuador.

La segunda base de datos, es la base final del proyecto, llamada 'Museo', esta llevará los datos definitivos resultado de los scripts de Migración realizados en la base intermedia 'MigracionQCA'.

Para realizar los Scripts de Migración, se hizo uso de los elementos del Lenguaje de Manipulación de Datos (DML) de SQL, la gran mayoría contenida en Stored Procedures para facilitar su ejecución.

### 4.3 Ejecución de la Migración de Información

La migración de la información se realizó por bloques, el primer bloque incluye la taxonomía principal, esto es: Reino, Filum, Clase, Orden, Familia, Género y Especie. Es en este punto donde se tuvo la principal complicación por los temas de información incompleta e inconsistente comentados anteriormente.

Para realizar la migración de elementos complicados que requieren de gran control, como es el caso de la taxonomía, se implementaron Cursores cargados con información de consultas de tablas temporales de la Base de datos intermedia 'MigracionQCA', para poder analizar la información de cada registro e ingresarlo en la categoría taxonómica que corresponda, para cada taxón que no tuviera información, fue ingresado un registro "No determinado" para lograr mantener la cadena taxonómica de las especies.

El segundo bloque fueron las Localidades, se utilizó una estrategia similar a la de la taxonomía para la migración de esta información, en gran parte por su característica jerárquica, partiendo de Continente hasta una localidad más particular. Igualmente, se ingresaron registros "No determinado" para datos no ingresados para mantener las conexiones.

El tercer bloque contiene a las subcategorías taxonómicas, esto incluyó a: Subfamilias, Tribus, Sub tribus, Subgéneros, Subespecies, variedades y formas, se realizó de esta manera para no interferir en la migración de la taxonomía principal, debido a que muchas de estas subcategorías no tenían conexión con su categoría más particular. Los taxones que, si tienen estas relaciones, fueron unidos con su categoría taxonómica correspondiente.

El cuarto bloque incluye a los especímenes, se ingresó la información de los especímenes, sus respectivas partes adicionales (denominados hijos), la información de su determinación, nombres comunes, información de herbarios y también se los relacionó con sus localidades, taxonomía.

El quinto bloque contiene a las listas de valores, información de colectores, información de préstamos, literatura, Información sobre Usos (Plantas útiles), Glosarios, Sinónimos, Proyectos, información de depuraciones, etc. Además de relacionar la información respectiva con los Especímenes y taxonomía.

La migración de la información toma alrededor de 15 horas en ejecutarse, principalmente debido al volumen de la información, ya que existen casi 220.000 especímenes almacenados, y alrededor de 33.000 taxones, hay que resaltar que las localidades también ocupan gran

cantidad de tiempo en migrarse, debido a que cada espécimen, suele contar con unas coordenadas de recolección específicas, lo que incurre que se inserte una localidad diferente así sea el mismo entorno que otra previamente ingresada, se registraron alrededor de 150.000 localidades diferentes.

Otro aspecto a tomar en cuenta es el uso de cursores en ciertos scripts de migración, esto ocasiona que la inserción de información se dé a velocidades más cortas, aunque la principal ventaja es que los registros van siendo analizados uno por uno y tratados de forma más controlada, lo que deriva en datos ingresados ordenadamente y con menores probabilidades de error.

Cada migración masiva de datos fue exhibida al personal del HerbarioQCA para su aprobación y revisión de posibles errores e inconsistencias.

#### 4.4 Verificación de la Migración y Consultas Complejas

FileMaker permite realizar consultas de acuerdo a cualquier criterio, lo cual lo convierte en una base de datos muy útil en este aspecto.

Para solucionar el problema de consultas complejas y específicas, se optó por usar una herramienta de inteligencia de negocios que permite, entre otras cosas, realizar análisis organizando la información en un Cubo Multidimensional (Hiper cubo), donde cada eje representa un criterio de análisis y cada cruce los valores a ser analizados.

Se utilizó la herramienta QlikView de la Empresa Qlik, que es una herramienta provisor de todas las funcionalidades necesarias para crear una interfaz de consultas con los campos necesarios para obtener la información deseada.

Adicionalmente, se puede exportar cada resultado obtenido a hojas de cálculo de Excel.

MUSEO DE ZOOLOGIA QCAZ

Buscar

Selección Actual  
 reino Plantae  
 clase Angiospermae

Limpia Selección

reino Animal  
 Plantae  
 Fungi

filum Anthocerotophyta  
 Arthropoda  
 Ascomycota  
 Basidiomycota  
 Charophyta  
 Chlorophyta  
 Chordata  
 Embryophyta  
 Marchantiophyta  
 Myxomycota  
 Phaeophyta  
 phylum desconocido  
 Rhodophyta

CLASES 1

orden 1

FAMILIAS 263

GENEROS 3035

ESPECIES 22314

TAXONOMIA

filum	clase	orden	familia	genero	especie
phylum desconocido	Angiospermae	orden desconocido	Rosaceae	Polylepis	(incana x sericea) x sericea
phylum desconocido	Angiospermae	orden desconocido	Solanaceae	Brugmansia	x rubella
phylum desconocido	Angiospermae	orden desconocido	Bromelaceae	Aechmea	abbreviata
phylum desconocido	Angiospermae	orden desconocido	Orchidaceae	Leparthopsis	abbreviata
phylum desconocido	Angiospermae	orden desconocido	Orchidaceae	Masdevalla	abbreviata
phylum desconocido	Angiospermae	orden desconocido	Melastomataceae	Miconia	abbreviata
phylum desconocido	Angiospermae	orden desconocido	ERICACEAE	Orthaea	abbreviata
phylum desconocido	Angiospermae	orden desconocido	Orchidaceae	Prosthechea	abbreviata
phylum desconocido	Angiospermae	orden desconocido	Orchidaceae	Trisetella	abbreviata
phylum desconocido	Angiospermae	orden desconocido	EUPHORBIACEAE	Chamaesyce	abditia
phylum desconocido	Angiospermae	orden desconocido	Rubiaceae	Psychotria	abditia
phylum desconocido	Angiospermae	orden desconocido	Orchidaceae	Stelis	abditia
phylum desconocido	Angiospermae	orden desconocido	Orchidaceae	Draconanthes	aberrans
phylum desconocido	Angiospermae	orden desconocido	Orchidaceae	Pleurothallis	aberrans
phylum desconocido	Angiospermae	orden desconocido	ERICACEAE	Psammisia	aberrans
phylum desconocido	Angiospermae	orden desconocido	ASTERACEAE	Diplostegium	abietinum
phylum desconocido	Angiospermae	orden desconocido	ERICACEAE	Asplenium	abietinum

Exportar

Ilustración 15 Interface Cubo Taxonomía 1

MUSEO DE ZOOLOGIA QCAZ

Buscar

Selección Actual  
 reino Plantae

Limpia Selección

reino Animal  
 Plantae  
 Fungi

filum Anthocerotophyta  
 Arthropoda  
 Ascomycota  
 Basidiomycota  
 Charophyta  
 Chlorophyta  
 Chordata  
 Embryophyta  
 Marchantiophyta  
 Myxomycota  
 Phaeophyta  
 phylum desconocido  
 Rhodophyta

CLASES 17

orden 36

FAMILIAS 378

GENEROS 3533

ESPECIES 27401

TAXONOMIA

filum	clase	orden	familia	genero	especie
Phaeophyta	Phaeophyceae	ALGAE	DESMARESTIACEAE	Desmarestia	aculeata
Chlorophyta	Ulvophyceae	ALGAE	CLADOPHORACEAE	Cladophora	aeruginosa
Chlorophyta	Ulvophyceae	ALGAE	DELESSERIACEAE	Membranoptera	alata
Chlorophyta	Ulvophyceae	ALGAE	GIGARTINACEAE	Chondrus	crispus
Phaeophyta	Phaeophyceae	ALGAE	CHORDARIAEAE	Leathesia	difformis
Phaeophyta	Phaeophyceae	ALGAE	LAMINARIACEAE	Laminaria	digitata
Phaeophyta	Phaeophyceae	ALGAE	SARGASSACEAE	Sargassum	ecuadoreanum
Chlorophyta	Ulvophyceae	ALGAE	CERAMIAEAE	Plumaria	elegans
Phaeophyta	Phaeophyceae	ALGAE	SCYTOSIPHONACEAE	Petalonia	fascia
Phaeophyta	Phaeophyceae	ALGAE	CHORDARIAEAE	Chordia	flagelliformis
Chlorophyta	Ulvophyceae	ALGAE	DUMONTIACEAE	Dumontia	incrassata
Phaeophyta	Phaeophyceae	ALGAE	SCYTOSIPHONACEAE	Scytosiphon	lomentaria
Chlorophyta	Ulvophyceae	ALGAE	RHODOMELACEAE	Polysiphonia	nigrescens
Chlorophyta	Ulvophyceae	ALGAE	CORALLINACEAE	Corallina	officialis
Chlorophyta	Ulvophyceae	ALGAE	CYSTOCLONACEAE	Cystodinium	purpurascens
Phaeophyta	Phaeophyceae	ALGAE	LAMINARIACEAE	Laminaria	saccharina
Phaeophyta	Phaeophyceae	ALGAE	ELMINACEAE	Elysiya	serotina

Exportar

Ilustración 16 Interface Cubo Taxonomía 2

REINO  
Animal Fungi  
Plantae

BUSQUEDA  
? digite un patron de busqueda

SELECCION ACTUAL  
reino qcazmemonico Plantae  
filum qca30 Chlorophyta  
clase qca30 Ulvophyceae

FILTROS  
TAXONOMIA LOCALIDAD TABLAS TIPO ESTADO

TIPOS PUBLICACION TEJIDO

TOTAL: 1

ESPECIMEN

qcaz	qcazmem...	Nro. Campo	Fec. Recoleccion	Fec. Registro	Fec. Ing. Coleccion	Fec. Identificación	Fec. Modificación
30	QCA30						

OBSERVACION  
General Notas Campo Inf. Ecologica Inf. Adicional Identificación Obs. Identificación Ident. Verbatim

INFORMACION ADICIONAL

Taxonomia		Personal	Co colector	Localidad	Conservacion	Obs. conservacion									
reino	Plantae	filum	Chlorophyta	clase	Ulvophyceae	orden	desconocido	familia	BIGNONIACEAE	genero	Bignonia	especie	aequinoctialis	idespecie	500469

Ilustración 17 Interface Búsqueda Cubo Especimen

REINO  
Animal Fungi  
Plantae

BUSQUEDA  
? digite un patron de busqueda

SELECCION ACTUAL  
reino Plantae  
filum Chlorophyta  
clase Ulvophyceae

FILTROS  
LOCALIDAD TABLAS TIPO ESTADO

TIPOS PUBLICACION TEJIDO

TOTAL: 15

ESPECIMEN

qcaz	qcazmemonico	Nro. Campo	Fec. Recoleccion	Fec. Registro	Fec. Ing. Coleccion	Fec. Identificación	Fec. Modificación
118	QCA118	-	-	-	-	-	-
131	QCA131	-	-	-	-	-	-
132	QCA132	-	-	-	-	-	-
133	QCA133	-	-	-	-	-	-
136	QCA136	-	-	-	-	-	-
137	QCA137	-	-	-	-	-	-
156	QCA156	-	-	-	-	-	-
172	QCA172	-	-	-	-	-	-
174	QCA174	-	-	-	-	-	-
178	QCA178	-	-	-	-	-	-
258	QCA258	-	-	-	-	-	-
724	QCA724	-	-	-	-	-	-
725	QCA725	-	-	-	-	-	-
213133	QCA213133	-	-	-	-	-	-

OBSERVACION  
General Notas Campo Inf. Ecologica Inf. Adicional Identificación Obs. Identificación Ident. Verbatim

INFORMACION ADICIONAL

Taxonomia		Personal	Co colector	Localidad	Conservacion	Obs. conservacion									
reino	Plantae	filum	Chlorophyta	clase	Ulvophyceae	orden	ALGAE	familia	CERAMIAACEAE	genero	Plumaria	especie	elegans	idespecie	507804
Plantae	Chlorophyta	Ulvophyceae	orden desconocido	CHLOROPHYTA	Enteromorpha	linza								513705	
Plantae	Chlorophyta	Ulvophyceae	orden desconocido	CHLOROPHYTA	Ulva	lactuca								512678	

Ilustración 18 Interface Búsqueda Cubo Especimen 2

**REINO**  
Animal Fungi  
Plantae

**BUSQUEDA**  
? digite un patron de busqueda

**SELECCION ACTUAL**  
provincia Amazonas

**FILTROS**  
TAXONOMIA  
TABLAS TIPO  
ESTADO

**ESPECIMEN**

qcaz	qcaznamenico	Nro. Campo	Fec. Recoleccion	Fec. Registro	Fec. Ing. Coleccion	Fec. Identificación	Fec. Modificación
14	QCA14	-	-	-	-	-	-
435	QCA435	-	-	-	-	-	-
447	QCA447	-	-	-	-	-	-
686	QCA686	-	-	-	-	-	-
882	QCA882	-	-	-	-	-	-
5405	QCA5405	-	-	-	-	-	-
5406	QCA5406	-	-	-	-	-	-
5420	QCA5420	-	-	-	-	-	-
5446	QCA5446	-	-	-	-	-	-
5450	QCA5450	-	-	-	-	-	-
5468	QCA5468	-	-	-	-	-	-
5610	QCA5610	-	-	-	-	-	-
5709	QCA5709	-	-	-	-	-	-
5813	QCA5813	-	-	-	-	-	-

**LOCALIDAD**  
PROVINCIA Amazonas  
CONTINENTE Oceania  
PAIS Acre

**OBSERVACION**  
General

**INFORMACION ADIC**  
Taxonomia Personal Co colector Localidad Conservacion Obs. conservacion

reino	filum	clase	orden	familia	genero	especie	idespecie
Plantae	phylum desco...	Angiospermae	orden desconocido	Annonaceae	Anavagorea	brevises	503436
Plantae	phylum desco...	Angiospermae	orden desconocido	Annonaceae	Anavagorea	dolichocarpa	507267
Plantae	phylum desco...	Angiospermae	orden desconocido	Annonaceae	Anavagorea	phaeocarpa	519050

**TIPOS** **PUBLICACION** **TEJIDO**

Ilustración 19 Interface Búsqueda Cubo Localidades

**BUSQUEDA**  
? digite un patron de busqueda

**SELECCION ACTUAL**  
especie nanay

**Categoría de Publicacion**  
Científica Divulgación

**Autor**  
A. Angulo  
A. Merino-Viteri  
B. E. Young  
C. C. Martinez  
C. J. Still

**Publicador**  
Surrey Beatty & Soons PTY Limited  
Abya-Yala  
American Museum of natural History  
American Ornithologists Union  
Asociación de Ecología y Conservación, f

**PUBLICACIONES**

**PUBLICACION**

Año	Titulo
2015	IUCN. 2010. The IUCN red list of threatened species
2014	Biogeographic analysis reveals ancient continental vicariance and recent oceanic dispersal in amphibians
2011	Amphipus, género y eclosión de renacuajos del jambato negro del Cajas en el Centro de Conservación de Anfibios del Bosque de Mazán en Ecuador
2011	Biodiversity and Conservation Status of Ecuadorian Amphibians
2010	Genetic differentiation in the nearly extinct harlequin frogs (Bufonidae: <i>Atelopus</i>), with emphasis on the Andean <i>Atelopus ignescens</i> and <i>A. bomolochos</i>
2008	Threatened Amphibians of the World
2006	Widespread amphibian extinctions from epidemic disease driven by global warming
2005	Catastrophic population declines and extinctions in neotropical harlequin frogs (Bufonidae: <i>Atelopus</i>).
2005	Ranas arlequines.

**AUTOR**

nombreAutor	Año Nacimiento	Lugar Nacimiento	idAutor
A. Angulo	-	-	501
A. Merino-Viteri	0	-	207
B. E. Young	-	-	349
C. C. Martinez	0	-	4931
C. J. Still	-	-	309
Diego Alvarado	-	-	13
E. La Marca	-	-	169

**Datos Gene...** **Intereses I...** **Experiencia** **Posicion A...** **Premios** **Asociacion...**

Ilustración 20 Interface Búsqueda Cubo Publicaciones

## Capítulo 5

### 1. Conclusiones:

- El desarrollo del producto de software se vio potenciado gracias a la utilización de conceptos de la ingeniería de Software, al ser un mecanismo de apoyo para el análisis y diseño basándose en buenas prácticas estandarizadas y documentadas que, al ser aplicadas correctamente, garantizan buenos resultados a lo largo del desarrollo del proyecto.
- La utilización de una metodología de desarrollo ágil como SCRUM, y la combinación en su etapa inicial con un estándar, como el IEEE 830, para la definición de requerimientos funcionales y no funcionales del software, definió un marco de trabajo dinámico y rápido que, desde el primer momento, permitió obtener resultados y optimizar el tiempo. Adicionalmente, ayudó a crear un producto que cumple con las expectativas y necesidades del cliente gracias a la retroalimentación periódica recibida en las reuniones perpetradas.
- El proceso de diseño de base de datos, al ser considerado como el corazón del sistema, requirió un tiempo considerable para su elaboración. La correcta definición del modelo permitió realizar la migración de la base de datos anterior, localizada en el sistema FileMaker, a SQL Server de manera organizada. Además, el desarrollo de las consultas resultó más sencillo y los resultados pueden ser obtenidos en tiempos cortos.
- La evolución constante de la tecnología, permitió que la implementación de software sea independiente de la plataforma, al adaptar diferentes tecnologías en el desarrollo para cumplir con los requerimientos planteados, aprovechando las capacidades de ASP.NET Web API y AngularJS.
- Las pruebas parciales realizadas durante el desarrollo de cada módulo, agilitaron el proceso de mejora, al no ser necesario tener en funcionamiento toda la aplicación, diagnosticando de manera temprana que componentes presentan problemas para su corrección.
- El sistema web desarrollado, ofrece una herramienta eficaz pensada en reducir los problemas de ingreso, manipulación, organización y seguridad de la información; Los 3 primeros son logrados mediante el uso de interfaces sencillas, minimalistas e intuitivas para los usuarios, mientras que la seguridad es conseguida gracias a la

implementación de técnicas refinadas de autenticación y acceso por roles a las funcionalidades del sistema.

- El desarrollo del sistema web, permitirá que en un futuro se pueda utilizar esta herramienta para administrar las colecciones desde cualquier parte del mundo y, mediante cualquier dispositivo que tenga acceso a la red, agilizar el proceso de digitalización y tratamiento de los datos.
- La implementación de un sistema adicional para reportes utilizando un cubo multidimensional, agrega valor al proyecto y mejora la productividad de la institución al permitir hacer el análisis de los datos científicos por medio de consultas complejas cuyos resultados son obtenidos en tiempos cortos que además admiten ser exportados de acuerdo a las necesidades de los usuarios.
- Los conocimientos y destrezas adquiridas al cursar la carrera de Ingeniería de Sistemas y Computación servirán de base para el desenvolvimiento de nuestras vidas profesionales, ya que el mundo en que nos desenvolvemos es muy dinámico y exige estar preparados para cualquier situación.
- La PUCE es una institución de larga trayectoria que ha permitido nuestro desarrollo como nuevos profesionales con valores, para aportar al desarrollo de la sociedad.
- La sociedad ecuatoriana se adapta a las nuevas tecnologías conforme cambian con el tiempo, sin embargo, no existe una visión sobre este tipo de proyectos que podrían ser aprovechados para el desarrollo del país.

## 2. Recomendaciones:

- En toda metodología de desarrollo de software, incluyendo SCRUM, es necesario que el cliente participe activamente en la definición de requerimientos, revisión de prototipos y las posteriores pruebas para garantizar el cumplimiento de sus necesidades.
- Es recomendable hacer uso de herramientas de control de versiones, que permiten que el trabajo en equipo sea más dinámico, tener un control sobre el progreso del sistema mediante el número de versiones lo que permite volver a un estado anterior en caso de que la compilación presente tenga problemas.
- Al desarrollar un sistema web, debemos ser conscientes de la presencia de nuevos dispositivos que permiten el acceso al mismo. Por lo que el diseño de interfaces debe ser, en lo posible adaptable a diferentes tamaños de pantalla existentes.

- Antes de realizar un rediseño de una base de datos y la posterior migración de información, es importante conocer bien el funcionamiento del sistema anterior y las relaciones entre entidades para verificar si los datos del sistema anterior se adaptan al nuevo diseño.
- Previo al desarrollo del software es importante establecer estándares de codificación para facilitar el trabajo del equipo, mejorar la legibilidad y reciclaje de código para disminuir el tiempo de implementación.
- Para mejorar la adaptación de los usuarios en la utilización de un nuevo sistema, se debe procurar que las interfaces sean amigables y presenten similitudes con la ubicación de los campos presentes en el sistema anterior.
- La carrera de Ingeniería de Sistemas y Computación debería incentivar el desarrollo de proyectos de investigación, así como emprendimientos para que los estudiantes adquieran destrezas que puedan ser aplicadas en la vida laboral.
- La PUCE debe incentivar la practica antes que la teoría en sus carreras, para que los estudiantes estén mejor preparados para ambientes y situaciones reales.

## Anexos

Anexo 1 – Documento de Requerimientos SRS

Anexo 2 - Costos de Servicio Computación en la Nube

Anexo 3 – Diccionario de Datos

Anexo 4 – Diagramas de Secuencia y Colaboración

Anexo 5 – Manual de Usuario para el Herbario QCA

## Bibliografía

¿Qué es Azure? El mejor servicio en la nube de Microsoft | Microsoft Azure. (2017).

Azure.microsoft.com. Retrieved 13 March 2017, from <https://azure.microsoft.com/es-es/overview/what-is-azure/>

A. García-Beltrán, R. Martínez y J.A. Jaén. (2008). Fundamentos de programación.

07/03/2016, de Universidad Politécnica de Madrid Sitio web:

<http://ocw.upm.es/ciencia-de-la-computacion-e-inteligencia-artificial/fundamentos-programacion/contenidosteoricos/ocwfundamentosprogramaciontema1.pdf>

ADO.NET. (2017). Msdn.microsoft.com. Obtenido el 24/01/2017,

[https://msdn.microsoft.com/es-es/library/e80y5yhx\(v=vs.110\).aspx](https://msdn.microsoft.com/es-es/library/e80y5yhx(v=vs.110).aspx)

Analítica guiada | Software de Business Intelligence | QlikView. (2017). Qlik.com. Retrieved

20 January 2017, from <http://www.qlik.com/es-es/products/qlikview>

Andrade, E. (2017). Manual C#. Academia.edu. Retrieved 24 January 2017, from

[http://www.academia.edu/13258747/Manual\\_C\\_](http://www.academia.edu/13258747/Manual_C_)

AngularJS — Superheroic JavaScript MVW Framework. (2016). Angularjs.org. Retrieved 17

December 2016, from <https://angularjs.org/>

AngularJS. (2016). En.wikipedia.org. Retrieved 17 December 2016, from

<https://en.wikipedia.org/wiki/AngularJS>

Anónimo, ¿Qué es un Sistema Gestor de Bases de Datos o SGBD? (2007). Foro Tecnico.

Obtenido 17/12/2016, from <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>

Biblioteca de Microsoft SQL Server. (2016). Msdn.microsoft.com. Retrieved 6 March 2016,

from <https://msdn.microsoft.com/es-es/library/bb545450.aspx>

Biblioteca de Microsoft SQL Server. (2016). Msdn.microsoft.com. Obtenido 17/12/2016, from

<https://msdn.microsoft.com/es-es/library/bb545450.aspx>

- Bootstrap (front-end framework). (2016). En.wikipedia.org. Obtenido 17/12/2016,  
[https://en.wikipedia.org/wiki/Bootstrap\\_\(front-end\\_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))
- C Sharp. (2016). Es.wikipedia.org. Retrieved 6 March 2016, from  
[https://es.wikipedia.org/wiki/C\\_Sharp](https://es.wikipedia.org/wiki/C_Sharp)
- Cascading Style Sheets. (2016). En.wikipedia.org. Obtenido 17/12/2016,  
[https://en.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets)
- Cerón Martínez, C. (2015). Bases para el estudio de la flora ecuatoriana (p. 9). Quito:  
Editorial Universitaria.
- El nuevo PHP. Conceptos avanzados. (1st ed., pp. 77-78). España: Bubok Publishing S.L.
- Enciclopedia de las Plantas Útiles del Ecuador L. de la Torre, H. Navarrete, P. Muriel M., M.  
J. Macía & H. Balslev (eds.) Herbario QCA & Herbario AAU. Quito & Aarhus. 2008: 1-  
3
- Entity Framework. (2017). Msdn.microsoft.com. Obtenido el 24/01/2017,  
[https://msdn.microsoft.com/es-es/library/gg696172\(v=vs.103\).aspx](https://msdn.microsoft.com/es-es/library/gg696172(v=vs.103).aspx)
- Tapias, J et al. (2016). Formato IEEE 830. Slideshare.net. Retrieved 20 July 2016, from  
[https://www.slideshare.net/Juan\\_Tapias/formato-ieee830srs-llevo](https://www.slideshare.net/Juan_Tapias/formato-ieee830srs-llevo)
- Herbario QCA. (2015). Herbario, Información general. 29-02-2016, de Herbario QCA Sitio  
web:  
<http://www.puce.edu.ec/portal/content/Informaci%C3%B3n%20General/541?link=oln30.redirect>
- Hernández Camacho, J. (2009). Necesidades de y Justificaciones BD.
- HTML. (2016). Es.wikipedia.org. Obtenido el 6/03/2016, <https://es.wikipedia.org/wiki/HTML>

- IEEE SA - 830-1998 - IEEE Recommended Practice for Software Requirements Specifications. (2017). Standards.ieee.org. Retrieved 24 January 2017, from <https://standards.ieee.org/findstds/standard/830-1998.html>
- Información general sobre el archivo .edmx (Entity Framework). (2017). Msdn.microsoft.com. Retrieved 24 January 2017, from [https://msdn.microsoft.com/es-es/library/cc982042\(v=vs.100\).aspx](https://msdn.microsoft.com/es-es/library/cc982042(v=vs.100).aspx)
- Información General Herbario QCA. (2016). Puce.edu.ec. Retrieved 30 June 2016, from <http://www.puce.edu.ec/portal/content/Herbario/477?link=oln30.redirect>
- Inteligencia empresarial. (2017). Es.wikipedia.org. Retrieved 20 January 2017, from [https://es.wikipedia.org/wiki/Inteligencia\\_empresarial](https://es.wikipedia.org/wiki/Inteligencia_empresarial)
- Introducción a .NET Framework. (2017). Msdn.microsoft.com. Obtenido el 24/01/2017, [https://msdn.microsoft.com/es-es/library/hh425099\(v=vs.110\).aspx](https://msdn.microsoft.com/es-es/library/hh425099(v=vs.110).aspx)
- Introducción a las bases de datos según Korth. (2017). Scribd. Retrieved 25 January 2017, from <https://www.scribd.com/doc/30859379/Introduccion-a-las-base-de-datos-segun-Korth>
- J. Rumbaugh, I. Jacobson, G. Booch. (1999). El lenguaje Unificado de Modelado. Manual de Referencia. Madrid: Pearson Educación, S.A.p.3.
- JavaScript. (2016). Es.wikipedia.org. Obtenido 6/03/2016, <https://es.wikipedia.org/wiki/JavaScript>
- Joyanes, L. Fundamentos de programación, Algoritmos y Estructuras de Datos, McGrawHill, Segunda edición, 1996
- JSON Introduction. (2016). W3schools.com. Retrieved 17 December 2016, from [http://www.w3schools.com/js/js\\_json\\_intro.asp](http://www.w3schools.com/js/js_json_intro.asp)
- JSON. (2016). Json.org. Retrieved 17 December 2016, from <http://www.json.org/json-es.html>

Laboratorio Nacional de Calidad del Software de INTECO (2009) pg.39

Loshin, D. (2012). Business Intelligence: The Savvy Manager's Guide. Waltham, MA: Morgan Kaufmann.

Métodos Informáticos en TurboPascal, García-Beltrán, A., Martínez, R., Jaén, J.A., Editorial Bellisco, 2002.

Microsoft Visual Studio. (2017). Es.wikipedia.org. Obtenido 24/01/2017, [https://es.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://es.wikipedia.org/wiki/Microsoft_Visual_Studio)

Mockup. (2017). En.wikipedia.org. Retrieved 13 March 2017, from [https://en.wikipedia.org/wiki/Mockup#Systems\\_engineering](https://en.wikipedia.org/wiki/Mockup#Systems_engineering)

Model–view–controller. (2016). En.wikipedia.org. Retrieved 17 December 2016, from <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>

MVC Framework Introduction. (2017). www.tutorialspoint.com. Retrieved 24 January 2017, from [http://www.tutorialspoint.com/mvc\\_framework/mvc\\_framework\\_introduction.htm](http://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm)

OAF MVC Architecture. (2012). Dibyajyoti Koch: A Blog on Oracle Application. Retrieved 17 December 2016, from <https://imdjkoeh.wordpress.com/2012/09/12/oaf-mvc-architecture/>

Omar, N. (2017). AngularJS: Create a Web Site using AngularJS, ASP.NET MVC, Web API - PART 1: Introduction. Oniwebblog.blogspot.com. Retrieved 27 January 2017, from <http://oniwebblog.blogspot.com/2016/01/angularjs-create-web-site-using.html>

Online Mockup, Wireframe & UI Prototyping Tool · Moqups. (2017). Moqups.com. Retrieved 13 March 2017, from <https://moqups.com/>

Peña, A. (2006). Ingeniería de Software: Una Guía para Crear Sistemas de Información. Retrieved 6 March 2016, from [http://www.wolnm.org/apa/articulos/ingenieria\\_software.pdf](http://www.wolnm.org/apa/articulos/ingenieria_software.pdf)

Remis, A., Remis, A., Remis, A., & perfil, V. (2011). Scrum e Ingenieria Software. Scrumiss.blogspot.com. Retrieved 24 January 2017, from <http://scrumiss.blogspot.com/>

Chávez, A, & Tenorio, J. (2012). Desarrollo de un sistema de control escolar para la escuela bilingüe Sangay combinando las metodologías SCRUM y XTREME PROGRAMMING (tesis de pregrado). Escuela Politécnica del Ejército, Sangolquí, Ecuador. from <http://repositorio.espe.edu.ec/bitstream/21000/5719/1/T-ESPE-033705.pdf>

SCRUM Methodology in the web development industry. (2017). Brightlabs Digital Agency. Retrieved 14 March 2017, from <https://www.brightlabs.com.au/page/Blog/SCRUM-Methodology-for-Websites/>

Servicio web. (2016). Es.wikipedia.org. Retrieved 17 December 2016, from [https://es.wikipedia.org/wiki/Servicio\\_web](https://es.wikipedia.org/wiki/Servicio_web)

Software requirements specification. (2017). En.wikipedia.org. Retrieved 24 January 2017, from [https://en.wikipedia.org/wiki/Software\\_requirements\\_specification](https://en.wikipedia.org/wiki/Software_requirements_specification)

Sommerville, I., & Alfonso Galapienso, M. (2005). Ingeniería del software (7th ed., p. 6). Madrid: Pearson Addison-Wesley.

Transferencia de Estado Representacional. (2016). Es.wikipedia.org. Retrieved 17 December 2016, from [https://es.wikipedia.org/wiki/Transferencia\\_de\\_Estado\\_Representacional](https://es.wikipedia.org/wiki/Transferencia_de_Estado_Representacional)

What is REST (representational state transfer)? - Definition from WhatIs.com. (2016). SearchSOA. Retrieved 17 December 2016, from <http://searchsoa.techtarget.com/definition/REST>