



**UNIVERSIDAD DEL BÍO-BÍO**  
**FACULTAD DE CIENCIAS EMPRESARIALES**  
**INGENIERIA CIVIL EN INFORMATICA**

# **Desarrollo de una aplicación web que gestione una agenda personal flexible**

**Nombre:** César Edmundo Guzmán Cádiz

**Profesor Guía:** María Angélica Caro Gutiérrez

**Profesor Informante:** Claudio Andres Muñoz Sepúlveda

## Resumen

Este proyecto se genera a raíz de la necesidad de originar una idea para un nuevo proyecto título, que por supuesto surgió de una necesidad real de resolver un problema, el cual es una agenda web personal flexible. Una agenda es un Libro o cuaderno en que se apunta, para no olvidarlo aquello que se ha de hacer (Diccionario de la Real Academia de la Lengua Española), y uno de los usos más importante es la ayuda en gestionar algo tan valioso como el tiempo.

Ya con la elección de este nuevo desafío, se comienza estudiando y analizando lo que es una agenda, para qué sirve y qué contiene. Luego de resolver esta duda y entender la importancia de este cuaderno, el cual organiza nuestro valioso tiempo, sabemos los reales beneficios de crear una agenda, la que se adapte a nuestros tiempos y necesidades.

Este proyecto, concretamente, tiene como propósito la creación de una agenda web personal flexible, que satisfaga las necesidades del propietario mucho más allá que una agenda tradicional. Esta agenda tendrá la facultad de ingresar Eventos y Actividades, que tienen una serie de atributos los cuales ayudarán a tener una mayor gestión de esta, y además, se tendrá la opción de generar distintos reportes, los cuales serán gestionados según las necesidades del propietario.

Para llevar a cabo este proyecto, se seleccionó una metodología de desarrollo llamado "Cascada", la cual contiene distintas etapas, donde el inicio de cada una de ellas depende de la fase anterior:

El objetivo final al cual debemos llegar, es cumplir todos los requerimientos del propietario, y que este aprenda a utilizar esta agenda web de forma fácil y sencilla, con acceso rápido y en cualquier lugar. Que gestione sus datos de forma simple, sin la más mínima dificultad.

## Índice General

<b>1</b>	<b>INTRODUCCIÓN.....</b>	<b>9</b>
<b>2</b>	<b>DESCRIPCIÓN DE LA PROBLEMÁTICA .....</b>	<b>11</b>
2.1	DESCRIPCIÓN DE LA PROBLEMÁTICA .....	11
<b>3</b>	<b>DEFINICIÓN PROYECTO .....</b>	<b>12</b>
3.1	OBJETIVOS DEL PROYECTO .....	12
3.1.1	OBJETIVO GENERAL.....	12
3.1.2	OBJETIVOS ESPECÍFICOS .....	12
3.2	AMBIENTE DE INGENIERÍA DE SOFTWARE.....	12
3.2.1	METODOLOGÍA DE DESARROLLO: .....	12
3.2.2	PARADIGMA DE PROGRAMACIÓN:.....	14
3.2.3	ARQUITECTURA .....	15
3.2.4	PATRONES DE DISEÑO .....	17
3.2.5	NOTACIONES DE MODELADO .....	18
3.2.6	TECNOLOGÍAS.....	18
3.2.7	HERRAMIENTAS .....	21
3.3	DEFINICIONES, SIGLAS Y ABREVIACIONES .....	22
<b>4</b>	<b>ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE .....</b>	<b>24</b>
4.1	ALCANCES .....	24
4.2	DESCRIPCIÓN GLOBAL DE LA INTERFAZ DE USUARIO .....	25
4.3	OBJETIVO DEL SOFTWARE .....	25
4.3.1	OBJETIVO GENERAL .....	25
4.3.2	OBJETIVOS ESPECÍFICOS .....	25
4.4	REQUERIMIENTOS ESPECÍFICOS .....	26
4.4.1	REQUERIMIENTOS FUNCIONALES DEL SISTEMA.....	26
4.4.2	REQUERIMIENTOS NO FUNCIONALES DEL SISTEMA.....	28
4.4.3	INTERFACES EXTERNAS DE ENTRADA.....	29
4.4.4	INTERFACES EXTERNAS DE SALIDA.....	29
4.4.5	ATRIBUTOS DEL PRODUCTO .....	30
<b>5</b>	<b>FACTIBILIDAD.....</b>	<b>31</b>
5.1	FACTIBILIDAD TÉCNICA. ....	31
5.1.1	SOFTWARE .....	31
5.1.2	HARDWARE.....	32

5.1.3	CONCLUSIÓN FACTIBILIDAD TÉCNICA.....	33
<b>5.2</b>	<b>FACTIBILIDAD OPERATIVA. ....</b>	<b>33</b>
5.2.1	COMPLEJIDAD DEL USO .....	33
5.2.2	RESISTENCIA A LA IMPLEMENTACIÓN .....	33
5.2.3	ADAPTACIÓN .....	34
5.2.4	CONCLUSIÓN DE LA FACTIBILIDAD OPERATIVA .....	34
<b>6</b>	<b><u>ANÁLISIS .....</u></b>	<b><u>35</u></b>
<b>6.1</b>	<b>DIAGRAMA DE CASOS DE USO .....</b>	<b>35</b>
6.1.1	ACTORES.....	36
6.1.2	CASOS DE USO Y DESCRIPCIÓN .....	36
6.1.3	ESPECIFICACIÓN DE LOS CASOS DE USO .....	37
<b>6.2</b>	<b>MODELAMIENTO DE DATOS .....</b>	<b>49</b>
6.2.1	DESCRIPCIÓN DE ENTIDADES.....	49
<b>7</b>	<b><u>DISEÑO.....</u></b>	<b><u>51</u></b>
<b>7.1</b>	<b>DISEÑO FÍSICO DE LA BASE DE DATOS .....</b>	<b>51</b>
<b>7.2</b>	<b>DISEÑO INTERFAZ Y NAVEGACIÓN .....</b>	<b>52</b>
7.2.1	INTERFAZ PRINCIPAL .....	52
7.2.2	INTERFAZ ACCIONES .....	54
7.2.3	ESQUEMA DE NAVEGACIÓN .....	57
<b>7.3</b>	<b>DISEÑO GENERAL DE ARQUITECTURA FUNCIONAL .....</b>	<b>58</b>
<b>8</b>	<b><u>PRUEBAS:.....</u></b>	<b><u>60</u></b>
<b>8.1</b>	<b>ELEMENTOS DE PRUEBA.....</b>	<b>60</b>
<b>8.2</b>	<b>ESPECIFICACIÓN DE LAS PRUEBAS.....</b>	<b>61</b>
8.2.1	CREAR EVENTO .....	61
8.2.2	EDITAR EVENTO .....	62
8.2.3	ELIMINAR EVENTO .....	63
8.2.4	GENERAR REPORTE EVENTO.....	63
8.2.5	CREAR ACTIVIDAD .....	65
8.2.6	EDITAR ACTIVIDAD .....	66
8.2.7	GENERAR REPORTE ACTIVIDAD .....	67
8.2.8	EDITAR SUBACTIVIDAD .....	68
8.2.9	AGREGAR SUBACTIVIDAD .....	69
8.2.10	CREAR CATEGORÍA .....	70
8.2.11	EDITAR CATEGORÍA .....	71

8.2.12	ELIMINAR CATEGORÍA.....	72
8.2.13	INICIAR SESIÓN .....	73
8.2.14	CERRAR SESIÓN .....	74
8.2.15	COMPATIBILIDAD WEB .....	74
8.2.16	COMPATIBILIDAD RESPONSIVE .....	75
<b>8.3</b>	<b>RESPONSABLES DE LAS PRUEBAS.....</b>	<b>77</b>
<b>8.4</b>	<b>CONCLUSIONES DE PRUEBAS.....</b>	<b>77</b>
<b>9</b>	<b><u>FACILIDAD DE USO .....</u></b>	<b><u>78</u></b>
<b>9.1</b>	<b>DEFINICIÓN DE ETAPAS EVALUACIÓN .....</b>	<b>78</b>
9.1.1	ETAPA 1.....	78
9.1.2	ETAPA 2.....	79
9.1.3	ETAPA 3.....	79
9.1.4	ETAPA 4.....	79
9.1.5	ETAPA 5.....	81
<b>9.2</b>	<b>RESULTADOS.....</b>	<b>85</b>
9.2.1	ANÁLISIS 1 .....	86
9.2.2	ANÁLISIS 2 .....	87
9.2.3	ANÁLISIS 3 .....	88
9.2.4	ANÁLISIS 4.....	89
9.2.5	CONCLUSIÓN.....	89
<b>10</b>	<b><u>CONCLUSION .....</u></b>	<b><u>90</u></b>
<b>11</b>	<b><u>BIBLIOGRAFÍA .....</u></b>	<b><u>91</u></b>
<b>12</b>	<b><u>ANEXO1 : MANUAL DE USUARIO .....</u></b>	<b><u>92</u></b>
<b>13</b>	<b><u>ANEXO2 : CUESTIONARIO DE FACILIDAD DE USO .....</u></b>	<b><u>93</u></b>
<b>14</b>	<b><u>ANEXO3 : MANUAL DE RÉPLICA .....</u></b>	<b><u>94</u></b>

## Índice Tablas

Tabla 4.1: Requerimientos Funcionales del sistema .....	27
Tabla 4.2: Requerimientos no funcionales del sistema.....	29
Tabla 4.3: Interfaces externas de entrada .....	29
Tabla 4.4: Interfaces externas de Salida.....	30
Tabla 5.1: Software Computador de escritorio .....	31
Tabla 5.2: Software Dispositivos móviles.....	31
Tabla 5.3: Hardware Computador de escritorio.....	32
Tabla 5.4: Hardware Dispositivos móviles .....	32
Tabla 6.1: actores del sistema.....	36
Tabla 6.2: Casos de Uso .....	36
Tabla 6.3: Especificación de caso de uso Crear Evento .....	37
Tabla 6.4: Especificación de caso de uso Crear Actividad.....	38
Tabla 6.5: Especificación de caso de uso Editar categoría.....	39
Tabla 6.6: Especificación de caso de uso Crear Categoría.....	40
Tabla 6.7: Especificación de caso de uso Eliminar Evento .....	40
Tabla 6.8: Especificación de caso de uso Reporte Actividad .....	41
Tabla 6.9 :Especificación de caso de uso Eliminar Categoría.....	42
Tabla 6.10: Especificación de caso de uso Generar Reporte Eventos.....	42
Tabla 6.11: Especificación de caso de uso Editar Evento .....	43
Tabla 6.12: Especificación de caso de uso Editar Actividad Libre.....	44
Tabla 6.13: Crear cuenta.....	45
Tabla 6.14: Especificación de caso de uso Editar SubActividad .....	45
Tabla 6.15: Especificación Iniciar Sesión.....	46
Tabla 6.16: Especificación Cerrar Sesión.....	47
Tabla 8.1: Pruebas Crear Evento .....	61
Tabla 8.2: Prueba Editar Evento.....	62
Tabla 8.3: Eliminar Evento .....	63
Tabla 8.4: Prueba Generar Reporte Evento.....	64
Tabla 8.5: Prueba Crear Actividad.....	65
Tabla 8.6: Editar Actividad .....	66

Tabla 8.7:Prueba Generar Reporte Actividad .....	67
Tabla 8.8: Prueba Editar SubActividad.....	69
Tabla 8.9:Agregar SubActividad.....	70
Tabla 8.10:Prueba Crear Categoría.....	70
Tabla 8.11: Prueba Editar Categoría .....	71
Tabla 8.12: Prueba Eliminar Categoría.....	72
Tabla 8.13: Prueba Iniciar Sesión .....	73
Tabla 8.14:Prueba Cerrar Sesión .....	74
Tabla 8.15:Compatibilidad WEB.....	75
Tabla 8.16:Compatibilidad Responsive.....	76
Tabla 9.1:Interpretación de cada rango .....	81
Tabla 9.2:Puntuación elemental(%) .....	82
Tabla 9.3:Resultados de los cuestionarios .....	85
Tabla 9.4:datos normalizados a puntuación elemental .....	86
Tabla 9.5:puntuaciones elementales de categorías por participante.....	87
Tabla 9.6:puntuaciones elementales por categoría .....	87
Tabla 9.7:puntuaciones elementales por participante.....	88
Tabla 9.8:grado porcentual de conformidad global.....	89

## Índice Figuras

Figuras 3.2:Etapas de metodología de desarrollo cascada.....	14
Figuras 3.3:Diseño Arquitectónico Modelo Vista Controlador.....	16
Figuras 6.1:Diagrama de casos de uso .....	35
Figuras 6.2:Modelo Entidad Relación .....	49
Figuras 7.1:Modelo Relacional de base de datos .....	51
Figuras 7.2:Diseño interfaz y navegación.....	52
Figuras 7.3:Área Encabezado.....	53
Figuras 7.4:Área Menú.....	53
Figuras 7.5:Sección Eventos.....	53
Figuras 7.6:Sección Actividad .....	54
Figuras 7.7:Iniciar Sesión.....	54
Figuras 7.8:Crear Evento y Actividad.....	55
Figuras 7.9:Categoría .....	55
Figuras 7.10:Reporte Evento .....	56
Figuras 7.11:Reporte Actividad .....	56
Figuras 7.12:Esquema de navegación.....	57
Figuras 7.13:esquema simple de arquitectura MVC.....	58
Figuras 7.14:Componentes de la arquitectura del sistema .....	59
Figuras 9.1:criterios de aceptabilidad .....	82
Figuras 9.2:Grafico: Total por categoría.....	88

---

## 1 INTRODUCCIÓN

---

Este informe tiene como propósito exponer el desarrollo del proyecto "Aplicación web que gestione una agenda personal flexible". Una agenda es un libro o cuaderno en que se apunta, para no olvidarlo, aquello que se ha de hacer (Diccionario de la Real Academia de la Lengua Española).

Desde tiempos antiguos el hombre ha tenido la necesidad de anotar cosas importantes para que no se olviden en el tiempo, he aquí el uso y utilidad de las agendas. Generalmente las agendas son cuadernos en blanco, que en la mayoría de los casos están distribuidas de forma calendarizada, las hay de distintas formas y tamaños, unas grandes que tienen más espacio para las anotaciones y suelen tener información extra generalmente útil, como por ejemplo números de emergencia, fórmulas matemáticas, reglamentos de instituciones, hojas para apuntes, etc. También hay agendas pequeñas de bolsillo, las cuales son útiles para apuntes rápidos que pueden ser anotados en cualquier momento y lugar.

En la actualidad gracias a las nuevas tecnologías, todo se maneja digitalmente, debido a la facilidad de acceso y rapidez de estas. En consecuencia, resulta atractiva la posibilidad de contar con una agenda personal web, la cual ayudará a organizar actividades de forma fácil y sencilla. Una de las cosas que buscan las nuevas tecnologías es hacer más fácil la vida de las personas, por lo tanto, uno de los requisitos mínimos del proyecto es que la funcionalidad sea lo más sencilla posible, ya sea al ingresar un nuevo evento o actividad, o que tenga una interfaz intuitiva de fácil navegación, esto para que cualquier persona aprenda rápidamente a utilizarlo y administrar su propia agenda digital.

Este proyecto desarrollará una aplicación web de una agenda personal flexible, que satisfaga lo antes mencionado, y además todos los requerimientos que hagan al administrador de esta aplicación, cumplir con todas sus necesidades; estas directamente relacionadas con guardar y administrar sus eventos y actividades.

Por lo tanto, en este informe se registrará y documentará todo lo referente al desarrollo de la aplicación agenda web, se realizará una serie de estudios, los cuales ayudarán a un desarrollo más efectivo y nos lleve a un objetivo final esperado que cumpla con todos los requerimientos dados por el propietario.

En primer lugar se hará una descripción de la problemática, para así tener un conocimiento general y completo de lo que se está enfrentando, gracias a esto se podrá definir claramente el objetivo del proyecto y sus requerimientos finales.

Como en todo proyecto, se necesita organización y una serie de pasos a seguir para cumplir un objetivo, se implementa una metodología de desarrollo que nos garantizará un marco o guía de trabajo usado para planificar, controlar y estructurar todo el desarrollo de este proyecto, el cual es la metodología de cascada, que en este mismo informe se explicara con mayor detalle.

Por lo tanto, este informe nos expondrá cada etapa enfrentada en el proceso de desarrollo de este proyecto, para así tener como resultado una aplicación que cumpla con los requerimientos, que sea útil y que cumpla lo esperado por el propietario.

---

## 2 DESCRIPCIÓN DE LA PROBLEMÁTICA

---

### 2.1 Descripción de la problemática

Las personas siempre han tenido la necesidad de organizar su tiempo, por lo tanto, las agendas se han utilizado desde tiempos antiguos, estas permiten recordar y planificar diversos eventos para así tener tiempo libre o de ocio, debido a la adecuada organización del tiempo.

La forma tradicional de las agendas es similar a un cuadernillo, con distintos tamaños, ya sean grandes, de escritorio, de cartera o de bolsillo. En estas se puede escribir cualquier evento o actividad que el administrador estime conveniente, se puede escribir con cualquier tipo de lápiz. Las anotaciones pueden ser eventos calendarizados o actividades de forma libre, que por lo general pueden tener subactividades.

Los problemas que pueden tener las agendas tradicionales es que al ser de papel, estas pueden perderse o estropearse, y si la encuentra alguien puede ver información personal que no se desea mostrar a nadie y utilizarla para perjudicar al propietario, otro problema que puede surgir es que si una persona tiene mucha información escrita, será difícil encontrar y organizar los eventos y actividades, ya que entre tantas hojas, la búsqueda se hace un poco tediosa.

Por lo tanto, al existir la tecnología digital, se ha buscado la solución de hacer una agenda personal web que satisfaga todas las necesidades del administrador de una agenda personal.

Sin embargo, es importante mencionar que aunque logremos adaptarnos al mundo de hoy, siempre surgirán algunas problemáticas, las cuales pueden ser, por ejemplo, que la tecnología digital necesite energía eléctrica, y que a falta de ella, pueda ocasionar el no acceso a la información o posiblemente pérdidas de esta, otro problema es la seguridad de nuestros datos, que al estar en un mundo interconectado pueden surgir agentes maliciosos que se aprovechen y traten de ingresar a lugares no autorizados para perjudicar a los demás.

---

## 3 DEFINICIÓN PROYECTO

---

### 3.1 Objetivos del proyecto

#### 3.1.1 Objetivo General

"Desarrollar una Aplicación web que gestione una agenda personal flexible de forma fácil y sencilla"

#### 3.1.2 Objetivos Específicos

- Almacenar información histórica de actividades y eventos.
- Registrar actividades y eventos de forma fácil y sencilla, ya sea de forma libre o asociada a un rango de fechas.
- Controlar desarrollo de las actividades y estados de cumplimiento.
- Buscar actividades y eventos de forma rápida y de acuerdo a criterios.
- Aplicar seguridad para que solo tenga acceso el *propietario*.
- Proveer una interfaz sencilla e intuitiva.

### 3.2 Ambiente de Ingeniería de Software

#### 3.2.1 Metodología de desarrollo:

La metodología de desarrollo utilizada en este proyecto es Cascada, esta metodología consiste en separar el desarrollo en distintas etapas, en las cuales el inicio de cada una de ellas depende de la fase anterior (Pressman S, 2002).

Esta metodología cuenta las siguientes etapas:

*Etapas de Definición De Requerimientos:*

Esta etapa es la más importante de esta metodología, ya que en esta se definen los servicios, objetivos y requerimientos del proyecto "La Agenda personal flexible", se discutirá claramente cada punto, y también se definirán las restricciones que se podrían presentar.

*Etapa de diseño:*

Luego de aclarar los requerimientos se debe pasar a un modelo más concreto, entre ellos: casos de uso, MER, modelo de datos, entre otros, por lo tanto, establece una arquitectura completa del sistema.

*Etapa de implementación:*

Al tener todas las herramientas a mano, solo queda programar, haciendo uso de prototipos y modelos, verificando que se cumplan todas las especificaciones.

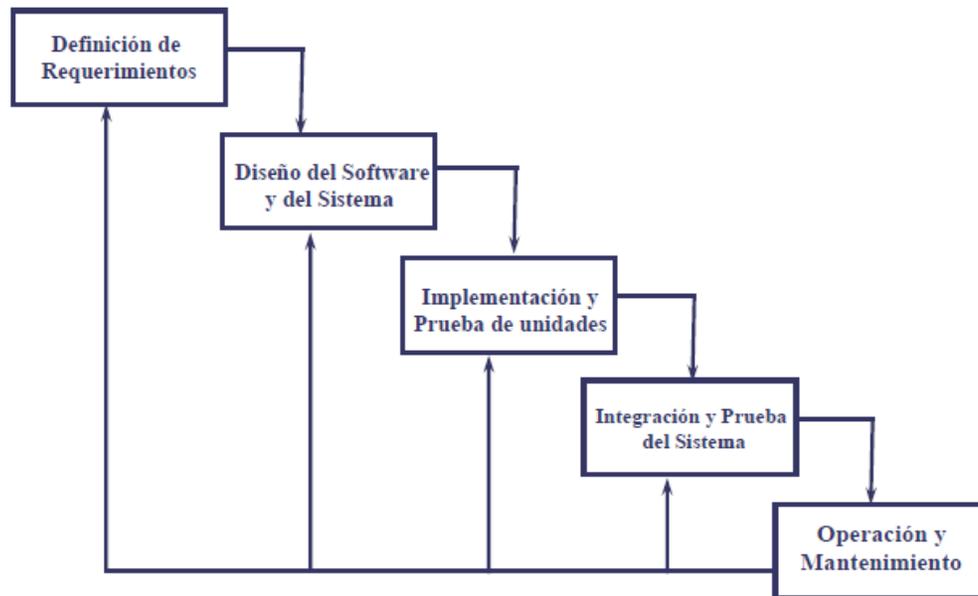
*Etapa integración y prueba:*

Se realizan las últimas pruebas de software y el manual de usuario para asegurar la mayor facilidad y eficacia en su implementación.

*Etapa de Operación:*

El sistema se instala y se pone en funcionamiento práctico.

Esta metodología es escalonada, y cada etapa es importante, puesto que si no se realiza correctamente, repercutirán en la etapa siguiente, es importante mencionar que los riesgos y errores siempre pueden estar presentes, no importando que metodologías se utilizan, sin embargo, se van haciendo menos recurrentes mientras mayor sea la experiencia. En esta metodología escalonada está presente la posibilidad de volver a una etapa anterior(ver figura 3.1 (Pressman S, 2002)), si se estima conveniente, algo primordial es darse cuenta lo antes posible de los errores, ya que mientras más se haya avanzado, los costos de repararlos serán aún más altos.



**Figuras 3.1: Etapas de metodología de desarrollo cascada**

### 3.2.2 Paradigma de Programación:

El paradigma de programación a utilizar en este proyecto es "La Programación Orientada a Objetos", este brinda un alcance favorable en diseño, desarrollo y mantención del software, ofreciendo soluciones de forma rápida a los problemas y preocupaciones que han existido desde el comienzo en el desarrollo de software: tales como la falta de portabilidad del código y reusabilidad, código que es difícil de modificar, ciclos de desarrollo largos y técnicas de codificación no intuitivas.

Su visión, es que los problemas a solucionar pueden ser representados con objetos, los cuales poseen un significado y comportamiento como los elementos del mundo real. Las características que tiene este paradigma es:

- Herencia: Donde una clase nueva se crea a partir de una clase existente, permite obtener características y comportamientos comunes desde un objeto con una morfología generalizada a partir de un conjunto de objetos similares.

- Encapsulamiento: Evita el acceso a datos por cualquier otro medio distinto a los especificados. Por lo tanto, la encapsulación garantiza la integridad de los datos que contiene un objeto.
- Modularidad: Permite modificar las características de las clases que definen a un objeto, de forma independiente a las demás clases dentro de la aplicación.
- Abstracción: Expresa las características esenciales de un objeto, las cuales distinguen al objeto de los demás.
- Polimorfismo: Capacidad que tienen los objetos de una clase de responder al mismo mensaje o evento en función de los parámetros utilizados durante su invocación. Un objeto polimórfico es una entidad que puede contener valores de diferentes tipos durante la ejecución del programa.

### 3.2.3 Arquitectura

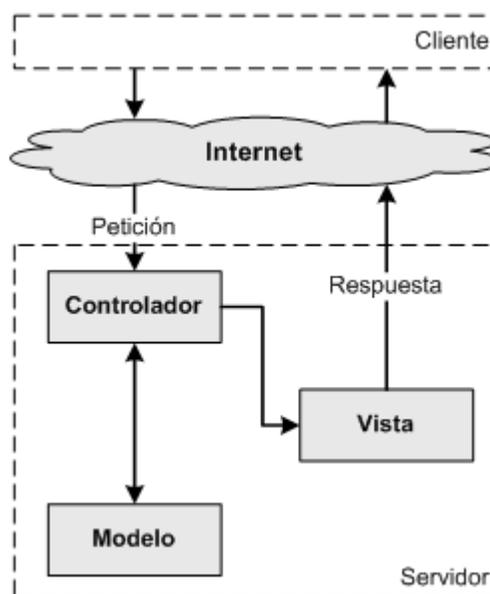
La arquitectura o patrón a utilizar es MVC (Modelo Vista Controlador), la cual su arquitectura está dividida en tres capas, los datos de la aplicación, interfaz de usuario y la lógica de control.

- Modelo: Contiene todo el contenido específico de la aplicación y la lógica de procesamiento, e incluye todos los objetos de contenido, el acceso a fuentes de datos externas y toda la funcionalidad de procesamiento que son específicos de la aplicación.
- Vista: Contiene todas las funciones específicas de la interfaz y habilita la presentación del contenido y la lógica de procesamiento, e incluye a todos los objetos de contenido, acceso a fuentes de datos externas y a toda la funcionalidad de procesamiento requerida por el usuario final.
- Controlador: Este responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista.

Algunas ventajas de este patrón:

- El diseño modular permite trabajar a diferentes desarrolladores en paralelo.
- Sus vistas muestran información actualizada siempre. El programador no debe preocuparse de solicitar que las vistas se actualicen, ya que este proceso es realizado automáticamente por el modelo de la aplicación.
- Podrás dividir la lógica de negocio del diseño, haciendo tu proyecto más escalable.
- Estabilidad en la aplicación debido al mejor control que tenemos sobre ella.
- Las modificaciones a las vistas no afectan al modelo de dominio, simplemente se modifica la representación de la información, no su tratamiento.

La figura 3.2 muestra gráficamente la arquitectura MVC.



Figuras 3.2: Diseño Arquitectónico Modelo Vista Controlador

Este modelo facilita algunas partes de la programación para hacerlas mucho más ágiles, al tener separado el código en diferentes capas, cuando se hacen modificaciones sobre la plantilla por ejemplo, solo hay que tocar esta parte que tiene un tipo de código más homogéneo y el resto no hace falta tocarlo para que funcione bien.

### 3.2.4 Patrones de diseño

Se define como “Soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos. Son soluciones basadas en la experiencia y se ha demostrado que funcionan”.

Razón de los patrones de diseño:

- Capturan la experiencia y la hacen accesible a los no expertos.
- Ayudan a la gente a comprender un sistema más rápidamente cuando está documentado con los patrones que usa.
- Aumentar la eficiencia en el desarrollo de un proyecto.
- El conjunto de sus nombres forma un vocabulario que ayuda a que los desarrolladores se comuniquen mejor
- Disminuir el esfuerzo de mantenimiento.

A continuación se detallan los patrones de diseño a utilizar en el desarrollo de este proyecto:

- Singleton: El patrón Singleton garantiza que una clase solo tenga una instancia y proporciona un punto de acceso global a ésta instancia. Se utiliza cuando existen varios clientes distintos que precisan referenciar a un mismo elemento, en donde se pretende asegurar que no hay más de una instancia de ese elemento.
- Date Transfer Object (DTO): Permite crear objetos que llevan todos los datos entre procesos o funciones, con el fin de reducir el número de llamadas de funciones para rescatar datos.
- Data Access Object (DAO): Es un patrón de diseño, que permite abstraer y encapsular todos los accesos a una base de datos en un objeto especial, que utiliza las conexiones para ejecutar las consultas necesarias para obtener y almacenar datos.

### **3.2.5 Notaciones de Modelado**

#### **3.2.5.1 UML (UNIFIED MODELING LANGUAGE)**

Lenguaje Unificado de Modelado (LUM o UML, por sus siglas en inglés, Unified Modeling Language), es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como, procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

Es importante remarcar que UML (fowler) es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

### **3.2.6 Tecnologías**

#### **3.2.6.1 Java**

Java es una tecnología que se usa para el desarrollo de aplicaciones (Oracle, 2014) que convierten a la Web en un elemento más interesante y útil. Java no es lo mismo que javascript, que se trata de una tecnología sencilla que se usa para crear páginas web y solamente se ejecuta en el explorador.

Java es un lenguaje de programación con el que se puede realizar cualquier tipo de programa, está centrado en el paradigma de orientación a objetos que es utilizado en la mayoría de los computadores del mundo. Sus aplicaciones son multiplataforma, es decir puede ser ejecutada en cualquier sistema operativo.

### **3.2.6.2 J2EE**

J2EE es el acrónimo de Java 2 Enterprise Edition, que es una plataforma de programación para desarrollar y ejecutar aplicaciones en java, principalmente enfocado al desarrollo de aplicaciones empresariales sobre interfaces web. Permite crear aplicaciones con capas y se apoya ampliamente en librerías y componentes de software modulares, ejecutándose sobre servidores web.

### **3.2.6.3 JSP**

JSP (Java Server Pages), que en castellano significa Páginas de Servidor Java. Proporciona una forma simplificada y rápida de crear contenido web dinámico, la tecnología JSP permite el rápido desarrollo de aplicaciones basadas en la web que son de servidor y de plataforma independiente. Posee diferentes etiquetas personalizadas que hacen interactuar los elementos de la página con objetos o datos del servidor

### **3.2.6.4 MySQL**

MySQL es el servidor de bases de datos relacionales más popular en el desarrollo de aplicaciones, gratuito y Open Source, este fue desarrollado y proporcionado por MySQL AB. MySQL AB es una empresa, cuyo negocio consiste en proporcionar servicios en torno al servidor de bases de datos MySQL.

### **3.2.6.5 Java Server Faces (JSF)**

Java Server Faces es un framework de componentes para la interfaz del usuario que funciona por el lado del servidor y se utiliza para el desarrollo de aplicaciones web basadas en la tecnología de Java.

### **3.2.6.6 CSS**

CSS es el acrónimo de “Cascading Style Sheets” (w3c, 2014), es un mecanismo que describe la forma en que se va a mostrar un documento en pantalla. De esta forma, el desarrollador cuenta con el control total sobre los formatos y el estilo del documento.

### 3.2.6.7 PrimeFaces

PrimeFaces es una librería de componentes visuales open source desarrollada y mantenida por Prime Technology (Çivici, 2014), una compañía Turca de IT especializada en consultoría ágil, JSF, Java EE y Outsourcing. El proyecto es liderado por Çağatay Çivici, un miembro del "JSF Expert Group".

Las principales características de Primefaces son:

- Soporte nativo de Ajax, incluyendo Push/Comet.
- Es un proyecto open source, activo y bastante estable entre versiones.
- kit para crear aplicaciones web para móviles.
- Es compatible con otras librerías de componentes, como [JBoss RichFaces](#).
- Uso de javascript no intrusivo (no aparece en línea dentro de los elementos, sino dentro de un bloque <script>).

### 3.2.6.8 Apache Tomcat

Apache Tomcat (también llamado Jakarta Tomcat o simplemente Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems.

Tomcat es mantenido y desarrollado por miembros de la Apache Software Foundation y voluntarios independientes. Los usuarios disponen de libre acceso a su código fuente y a su forma binaria en los términos establecidos en la Apache Software License. Las primeras distribuciones de Tomcat fueron las versiones 3.0.x. Las versiones más recientes son las 8.x. (Apache-Tomcat-Project, 2012)

## **3.2.7 Herramientas**

### **3.2.7.1 Netbeans IDE**

NetBeansIDE (Corporation, 2014)permite rápida y fácilmente desarrollar Java de escritorio, móviles y aplicaciones web, así como aplicaciones HTML5conHTML, JavaScript y CSS. ElIDE también proporciona un gran conjunto de herramientas para PHP y C / C ++desarrolladores de C. Es gratuito y de código abierto y tiene una gran comunidad de usuarios y desarrolladores de todo el mundo.

### **3.2.7.2 yEd Graph Editor**

Editor gráfico de una gran potencia, que entrega la posibilidad de crear de manera rápida esquemas o mapas conceptuales.

### **3.2.7.3 notepad++**

Editor de código fuente y un sustituto de Notepad (notepad, 2014)que soporta varios lenguajes. Funciona en entorno MS Windows y su uso está regido por la GPL License.

Basado en el potente componente de edición Scintilla , Notepad + + está escrito en C + + y utiliza la API de Win32 y STL, lo que asegura una velocidad mayor de ejecución y menor tamaño del programa. Mediante la optimización de las rutinas de las muchas posibles, sin perder la facilidad de uso, Notepad + + intenta reducir las emisiones mundiales de dióxido de carbono. Al utilizar menos energía de la CPU, el PC puede moderar la marcha y reducir el consumo de energía, lo que resulta en un ambiente más ecológico.

#### **3.2.7.4 PhpMyAdmin**

Herramienta escrita en php, cuya funcionalidad es la creación y administración de la base de datos, permitiendo así poder trabajar los datos ingresados en el sitio WEB de manera efectiva y eficiente.

### **3.3 Definiciones, Siglas y Abreviaciones**

Este ítem incluye las definiciones de las siglas, abreviaciones, conceptos técnicos o de negocio que son necesarios para comprender este documento.

Propietario:

-Que tiene derecho de propiedad sobre algo, y especialmente sobre bienes inmuebles.

(Real Academia Española)

-En el contexto del proyecto es el administrador de la aplicación.

Evento:

-Suceso importante y programado, de índole social, académica, artística o deportiva.

(Real Academia Española)

-Un evento es Calendarizado, y posee un título, rangos de tiempo y una categoría.

Actividad:

-Conjunto de operaciones o tareas propias de una persona o entidad.

(Real Academia Española)

-Una actividad es ingresada libre y consta de un título, categoría, estado, esta puede tener o no asociada distintas SubActividades.

SubActividad:

Una SubActividad está asociada a una actividad

**Categoría:**

-Uno de los diferentes elementos de clasificación que suelen emplearse en las ciencias.

(Real Academia Española)

-Una categoría es un distintivo o clasificador que poseerá una actividad o evento.

**Estado:**

-Situación en que se encuentra alguien o algo, y en especial cada uno de sus sucesivos modos de ser o estar.

(Real Academia Española)

-Un estado es dado a una actividad o evento y para representar el cumplimiento de esta.

**logueo:**

-Es un ámbito de seguridad informático, es un uso de la palabra login (log in), logear o logearse. Significa registrarse o identificarse para ingresar a un sitio restringido.

**Reporte:**

Un reporte permite tener una vista general o particular de los eventos o actividades, esto dependiendo de distintas filtros que el propietario estime conveniente.

---

## 4 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

---

### 4.1 Alcances

Este proyecto tiene como objetivo hacer una Aplicación web que gestione una agenda personal de forma fácil y flexible, este producto tendrá la facultad de ayudar a su propietario a organizar sus eventos calendarizados y actividades libres.

*Beneficios del proyecto:*

- Se tendrá un mayor acceso a los datos, ya que se podrá utilizar en cualquier parte con acceso a internet.
- El propietario tendrá mayor seguridad en sus datos, debido que la aplicación solo tendrá acceso ingresando nombre de usuario y contraseña.
- El propietario tendrá mayor eficiencia en la organización de sus actividades y eventos, puesto que la aplicación ayudará con sus distintas funcionalidades.
- Se llevará un control de todos los eventos y actividades por medio de distintos informes.

*Algunas limitantes son:*

- El sistema solo será operado por un propietario.
- Los informes solo serán mostrados por pantalla.
- Si el administrador se olvida de su clave tendrá que seguir un protocolo para recuperarla o cambiarla.

## **4.2 Descripción Global de la interfaz de usuario**

La interfaz de usuario tiene como característica ser sencilla e intuitiva, con tipografías entendibles y colores sobrios, provocando un fácil entendimiento, conocimiento y aprendizaje por parte del propietario.

## **4.3 Objetivo del software**

### **4.3.1 Objetivo general**

La aplicación tiene como objetivo un sistema que ayude al propietario a gestionar y administrar una agenda personal web, la cual permitirá el ingreso de eventos calendarizados y actividades libres, brindará organización debido a distintos criterios tales como fechas, estados o categorías, los cuales el sistema permitirá crearlos de forma personalizadas, además de tener la facultad de generar distintos reportes.

### **4.3.2 Objetivos Específicos**

- Permitir el registro de eventos calendarizados
- Permitir el registro de actividades libres.
- Permitir el registro de categoría.
- Generar distintos reportes para la mejor organización de eventos y actividades.
- Manejar los eventos calendarizados más próximos para que se muestren al propietario al cerrar sesión.
- Permitir acceso seguro a la información del propietario.

## 4.4 Requerimientos Específicos

### 4.4.1 Requerimientos Funcionales del sistema

A continuación se presentan en la tabla 4.1, los requerimientos funcionales del sistema

Id	Nombre	Descripción
RF01	Ingresar evento	El propietario podrá ingresar eventos especificando cuando se realizará, especificando rangos de tiempo, tendrá un título, una categoría y por defecto el evento quedará con estado "no realizado".
RF02	Ingresar actividad	El propietario podrá ingresar actividades de forma libre sin especificar tiempos, al ingresarlas tendrán un título, una categoría y un estado, esta podrá ser definida como cumplidas, no cumplidas o anuladas, y así también del mismo modo se le podrán agregar subActividades.
RF03	Buscar actividad o evento	El propietario podrá hacer búsquedas de actividades o eventos gracias a distintos filtros, ya sea por fecha o rangos de fechas, cumplidas o no cumplidas y categorías.
RF04	Iniciar Sesión	El ingreso de la aplicación debe ser gracias al nombre y clave únicos del propietario, para así luego el sistema ingrese a la aplicación.
RF05	Ingresar categorías	El propietario podrá ingresar nuevas categorías con el fin de asignarlas a un evento o actividad, y tener así criterios que ayuden a una mejor administración.
RF06	Generar reporte actividad	El propietario podrá visualizar un reporte en donde se verán de forma resumida el cumplimiento de las actividades,

		mostrándolas en una tablas, lo cual ayudará al propietario a ver qué tan alejado esta del cumplir sus metas.
RF07	Generar reporte evento	El propietario podrá visualizar un reporte en donde se verán de forma resumida el cumplimiento de los eventos, mostrándolas en una tablas, lo cual ayudará al propietario a ver qué tan alejado esta del cumplir sus metas.
RF08	Ver de forma destacada sus eventos más próximos	La aplicación tendrá la funcionalidad de mostrar los eventos más próximos de forma destacada al cerrar la sesión.
RF09	Editar Evento	El propietario tiene la opción de editar los eventos, ya sea fechas, título, estado o categoría.
RF10	Editar Actividad	El propietario tiene la opción de editar las actividades, ya sea título, categoría o sus sub actividades.
RF11	Editar categoría	El propietario posee la facultad de editar categorías, es decir permite cambiar el titulo de chicha categoría.
RF12	Eliminar categoría	El propietario tiene la facultad de eliminar una categoría que no esté asociada.
RF13	Serrar Sesión	El propietario podrá cerrar sesión en cualquier momento o que el sistema lo haga por inactividad del la aplicación.
RF14	Eliminar evento	El propietario tiene la facultad de eliminar un evento si así lo estima conveniente.
RF15	Crear cuenta	El propietario crean un cuanta antes de empezar a utilizar por primera vez la aplicación.

**Tabla 4.1: Requerimientos Funcionales del sistema**

#### 4.4.2 Requerimientos no funcionales del sistema

A continuación se presentan en la tabla 4.2, los requerimientos no funcionales del sistema

Id	Nombre	Descripción
RNF01	Proveer una interfaz sencilla e intuitiva	La aplicación debe tener un vista sencilla e intuitiva para que cualquier persona aprenda a utilizarla rápida y fácilmente, las opciones deben ser las justas y necesarias, con colores simples y agradables a la vista
RNF02	Poseer interfaces de tipo responsive	Hoy en día se tiene acceso a distintos dispositivos, ya sea móviles o de escritorio, y estas tienen distintas resoluciones, por lo tanto, hoy se utilizan vistas responsive, esto quiere decir un diseño web adaptable a las distintas plataformas, ya sea celular, tabletas o computadoras
RNF03	Compatibilidad Web	El sistema debe ser compatible con los navegadores más utilizados actualmente: Mozilla Firefox, Google Chrome, Internet Explorer.
RNF04	Visualización de Actividad y evento en una misma vista	Para mejorar el acceso de las actividades y eventos, se visualizarán ambas en la misma pantalla.

RNF05	Horarios visibles de eventos en rangos laborales.	Los horarios visibles para el propietario solo son útiles dentro el rango 8:00 a 20:00 horas, ya que la aplicación está enfocada para uso laboral.
-------	---	--

Tabla 4.2: Requerimientos no funcionales del sistema

#### 4.4.3 Interfaces externas de entrada

En la tabla 4.3 muestra cada interfaz de entrada e indica todos los grupos de datos que serán ingresados al sistema independiente del medio de ingreso.

Identificador	Nombre del ítem.	Detalle de Datos contenidos en ítem
IE_01	Datos del Evento	Título, Estado, Categoría, fecha, hora inicio, hora fin.
IE_02	Datos de la Actividad	Título, Estado, Categoría, subActividades.
IE_03	Datos de la Categoría	Nombre.
IE_04	Datos de nueva subActividad	Título, Estado.
IE_05	Dato de edición de SubActividad	Título, Estado.
IE_06	Dato de edición de Actividad	Título, Estado.
IE_07	Dato de edición de Evento	Título, Estado, Categoría, fecha, hora inicio, hora fin.
IE_08	Dato de edición de Categoría	Nombre.
IE_09	Datos de usuario	Nombre, contraseña.

Tabla 4.3: Interfaces externas de entrada

#### 4.4.4 Interfaces externas de Salida

En la tabla 4.4 se especifica cada salida del sistema, indicando en cada caso el formato o medio de salida.

Identificador	Nombre del ítem.	Detalle de Datos contenidos en ítem	Medio Salida
IS_01	Reporte Evento	Título, Fecha, Estado, Categoría	Pantalla
IS_02	Reporte Actividad	Título, Fecha, Estado, Categoría, subCategoría	Pantalla

IS_03	Lista de Actividades	Título, Estado	Pantalla
IS_04	Lista de Evento	Hora, Título	Pantalla
IS_05	Lista Categoría	Título	Pantalla
IS_06	Lista de Eventos próximos	Título, Fecha, Estado, Categoría	Pantalla
IS_07	Lista de SubActividades	Título, Estado	Pantalla

**Tabla 4.4: Interfaces externas de Salida**

#### **4.4.5 Atributos del producto**

##### **4.4.5.1 Funcionalidad**

- Seguridad: El sistema mantiene y exige al administrador autenticarse como tal, mediante una interfaz de inicio de sesión. Sin la autenticación no puede acceder a contenido de la agenda web. La sesión solo durara aproximadamente 30 minutos de inactividad y se cerrará.

##### **4.4.5.2 Usabilidad**

- Operabilidad: Los mensajes de error relacionados con ingresos y procesamiento de datos, indicaran claramente la causa, tipo de error y la solución de este de forma clara para el administrador

##### **4.4.5.3 Portabilidad**

- Adaptabilidad: El sistema podrá ser accesible de cualquier dispositivo, debido a que tendrá interface responsive (podrá ser visto sin problema en computadores, tabletas, smartphones)
- Accesibilidad: El sistema está en un servidor en la nube , por lo tanto, este se encuentra disponible en cualquier parte en la cual se tenga acceso a la web.

## 5 FACTIBILIDAD

### 5.1 Factibilidad técnica.

La factibilidad técnica estudia los aspectos tecnológicos y evalúa si los implementos necesarios ya sea software o hardware, se encuentran disponibles.

#### 5.1.1 Software

A continuación se explicitan algunos requerimientos mínimos de aplicaciones o programas necesarios para la correcta implementación del sistema, debido a que la aplicación será responsive, estará separado en computador de escritorio y dispositivos móviles (tablet, smartphone)

##### Computador de escritorio:

En la tabla 5.1 se muestran los requerimientos mínimos de sistema, aplicaciones o programas para un buen funcionamiento de la aplicación en un computador de escritorio.

Cantidad	Descripción
01	Sistema Operativo Windows xp,7,8
01	Navegador Web Google Chrome
01	Navegado Web Mozilla Firefox

Tabla 5.1: Software Computador de escritorio

##### Dispositivos móviles:

En la tabla 5.2 se muestran los requerimientos mínimos de sistema, aplicaciones o programas para un buen funcionamiento de la aplicación en un dispositivo móvil..

Cantidad	Descripción
01	Sistema operativo Android 2.X, 3.X, 4.X
01	Navegador móvil Safari para iOS
01	Navegador móvil Opera Mobile
01	Navegador móvil Firefox
01	sistema operativo Windows Phone
01	sistema operativo iOS 5
01	BlackBerry OS

Tabla 5.2: Software Dispositivos móviles

## 5.1.2 Hardware

En cuanto a hardware, se necesita que los equipos posean requisitos mínimos para el correcto funcionamiento de la aplicación.

### Computador de escritorio:

En la tabla 5.3 se muestran los requerimientos mínimos de una computadora para un buen funcionamiento de la aplicación.

Componente de Hardware	Requerimiento Mínimo
Procesador	Igual o superior a 1.5 GHz.
Disco Duro	Igual o superior a una capacidad de 20 Gb.
Memoria RAM	Igual o superior a 512 MB.
Tarjeta de Red	Cualquiera Compatible con Ethernet.
Tarjeta de Video	Que posea compatibilidad con VGA, o superior.
Monitor	Con resolución Variable
Teclado	Teclado tradicional.
Mouse	Mouse Tradicional o TouchPad.

Tabla 5.3: Hardware Computador de escritorio

### Dispositivos móviles:

En la tabla 5.4 se muestran los requerimientos mínimos móviles para un buen funcionamiento de la aplicación.

Componente de Hardware	Requerimiento Mínimo
Procesador	Igual o superior a 1 GHz.
Memoria	Igual o superior a una capacidad de 1 Gb.
Memoria RAM	Igual o superior a 512 MB.
Red	Conectividad Wi-Fi , 3G, 4G
Pantalla	Con resolución Variable
Teclado	Touch o Multi-Touch

Tabla 5.4: Hardware Dispositivos móviles

### **5.1.3 Conclusión Factibilidad técnica**

Se puede concluir en esta sección, que todas las tecnologías necesarias para que pueda implementarse el sistema, son accesibles por cualquier persona hoy en día, solo por mencionar en diciembre de 2012 habían 24,1 millones de celulares en Chile, un promedio de 1,38 por habitante, y de similar forma las computadoras ocupan un gran porcentaje de los hogares chilenos.

## **5.2 Factibilidad operativa.**

La factibilidad operativa determina la probabilidad de que un sistema se use como debería ser usado. Para este estudio se utilizarán 3 factores importantes:

- Complejidad del uso
- Resistencia a la implementación
- Adaptación

### **5.2.1 Complejidad del uso**

Los sistemas informáticos siempre tienen su complejidad, y parecen muy difíciles de utilizar, llevando a los usuarios a causar fallas o errores en el sistema.

En este caso, el proyecto se ha enfocado en que los procesos realizados sean lo más sencillo y estables posible, las funcionalidades serán las que el usuario ha indicado como requerimientos, es decir, el uso se adaptará más rápidamente al propietario de la agenda.

### **5.2.2 Resistencia a la implementación**

Siempre un nuevo sistema provoca resistencia a los usuarios o administrador al utilizarlo, ya que las personas están acostumbradas a lo tradicional, pero este proyecto tiene claros y sencillos requerimientos, por lo tanto, no tendría que haber tanta resistencia al implementar el sistema.

### **5.2.3 Adaptación**

La adaptación del sistema no debería ser mucha, ya que el procesos es el mismo que realiza el propietario de forma manual, además existirá un manual de usuario explicando clara y sencillamente cada uno de los pasos para administrar fácilmente la aplicación, el propietario participara activamente en el desarrollo de la aplicación, con opiniones y comentarios.

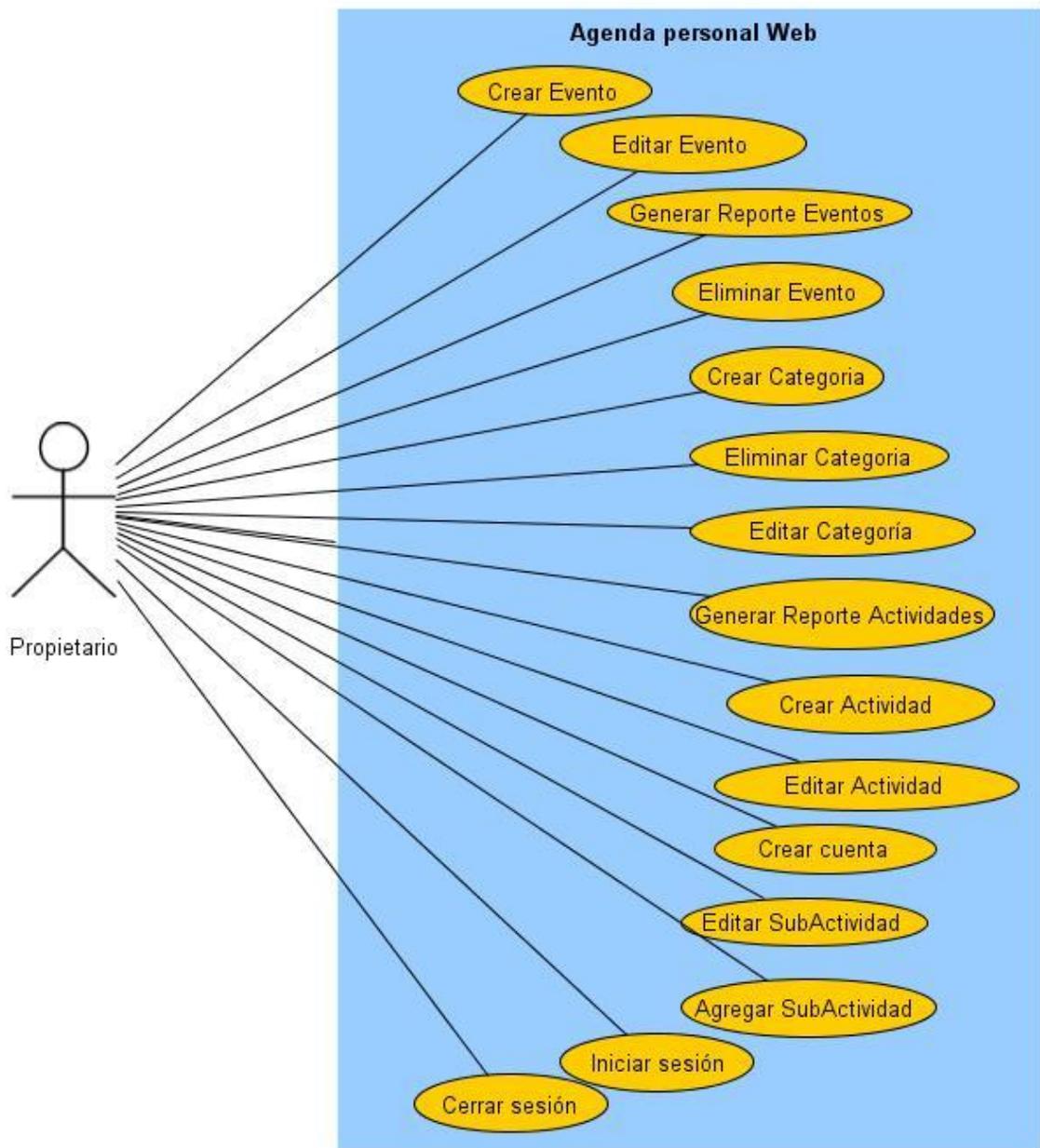
### **5.2.4 Conclusión de la factibilidad operativa**

Se puede concluir, que la factibilidad por la parte operativa no es tan compleja con respecto al proyecto tratado en este informe, ya que no tiene mayor complejidad de lo ya sabido por el administrador, pero como el sistema puede ser usado por cualquier persona existirá un manual de usuario explicando claramente todo lo respecto a la administración de la aplicación.

## 6 ANÁLISIS

### 6.1 Diagrama de casos de uso

La utilidad de este tipo de diagramas radica en que solo indica el “comportamiento” que se desea del sistema, por lo tanto, es un buen punto de partida para el análisis a desarrollar. La figura 6.1 muestra el diagrama de caso de uso.



Figuras 6.1:Diagrama de casos de uso

### 6.1.1 Actores

La tabla 6.1 expone los actores involucrados en el sistema.

ID	ACTOR	DESCRIPCION
A_01	Propietario	Persona que organice eventos/actividad personales

Tabla 6.1: actores del sistema

### 6.1.2 Casos de Uso y descripción

A continuación se presenta el diagrama de los casos de uso (tabla 6.2) para mostrar la interacción entre los actores y los casos de uso. Además del diagrama, se debe describir en forma textual (palabras) la forma como los actores harán uso del sistema. Esta descripción es general y no debe indicar el detalle de la ejecución de cada caso de uso.

Identificador	Nombre	ID actor
CU_01	Crear Evento	A_01
CU_02	Crear Actividad	A_01
CU_03	Editar Categoría	A_01
CU_04	Crear Categoría	A_01
CU_05	Eliminar Evento	A_01
CU_06	Generar Reporte Actividades	A_01
CU_07	Eliminar Categoría	A_01
CU_08	Generar Reporte eventos	A_01
CU_09	Editar Evento	A_01
CU_10	Editar Actividad	A_01
CU_11	Crear cuenta	A_01
CU_12	Editar Sub Actividad	A_01
CU_13	Iniciar Sesión	A_01
CU_14	Cerrar Sesión	A_01
CU_15	Agregar subActividad	A_01

Tabla 6.2: Casos de Uso

### 6.1.3 Especificación de los Casos de Uso

Las siguientes tablas mostraran las especificaciones de los casos de uso.

#### 6.1.3.1 Caso de Uso: Crear Evento

Caso de uso		Crear Evento	
ID	CU_01		
Descripción	El Propietario crea un Evento		
Pre-Condiciones	Haber iniciado sesión.		
Actores	<b>Propietario</b>		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario cliquee en alguna parte de la zona de los eventos.  3. El Propietario ingresa los datos requeridos y presiona la opción guardar.	2.El sistema muestra una interfaz donde se crea el evento.  4 .El sistema guarda los datos del evento y direcciona a interface principal.	
Flujo alternativo 1	El Propietario se arrepiente y presiona eliminar o cerrar(x) en la esquina superior derecha.	El sistema direcciona a la interface principal.	
Post condiciones	El sistema agregará el nuevo evento a la lista de eventos.		

Tabla 6.3: Especificación de caso de uso Crear Evento

**6.1.3.2 Caso de Uso: Crear Actividad**

Caso de uso		Crear Actividad	
ID	CU_02		
Descripción	El Propietario crea un Actividad		
Pre-Condiciones	Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario clikea la opción crear actividad(+)  3. El Propietario ingresa los datos requeridos y presiona la opción guardar.	2. El sistema le muestra una interfaz donde se crea la actividad.  4. El sistema guarda los datos de la actividad y direcciona a la interface principal.	
Flujo alternativo 1	El Propietario se arrepiente y presiona cerrar(x) en la esquina superior derecha.	El sistema direccionar a interface principal.	
Post condiciones	El sistema agregará una nueva Actividad a la lista de Actividades.		

Tabla 6.4: Especificación de caso de uso Crear Actividad

**6.1.3.3 Caso de Uso: Editar Categoría**

Caso de uso		Editar Categoría	
ID	CU_03		
Descripción	El Propietario Edita Categoría		
Pre-Condiciones	Haber iniciado sesión Debe haber alguna categoría ingresada		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario selecciona la opción Categoría	2. El sistema muestra una interfaz de Categoría.	

	<p>3. El Propietario edita la Categoría que estime conveniente presionado el botón editar  y luego presiona el botón guardar .</p>	<p>4.El sistema procesa los datos para editar la Categoría en la base de datos.</p>
Flujo alternativo 1	<p>3. El Propietario edita la Categoría que estime conveniente presionando el botón editar , luego el propietario presiona el botón cancelar  para no realizar cambios.</p>	<p>4. El sistema no realiza los cambios.</p>
Post condiciones	<p>El sistema editara la categoría en la lista.</p>	

Tabla 6.5: Especificación de caso de uso Editar categoría

#### 6.1.3.4 Caso de Uso: Crear Categoría

Caso de uso		Crear Categoría	
ID	CU_04		
Descripción	El Propietario Crear una Categoría		
Pre-Condiciones	Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario presiona la opción crear Categoría	2. El sistema le muestra una interfaz para crear la categoría.	

	3. El Propietario escribe el nombre de la categoría y presiona agregar.	4.EL sistema agrega la nueva categoría.
Flujo alternativo 1		
Post condiciones	El sistema agrega la nueva categoría a la lista.	

Tabla 6.6: Especificación de caso de uso Crear Categoría

### 6.1.3.5 Caso de Uso: Eliminar Evento

Caso de uso		Eliminar Evento	
ID	CU_05		
Descripción	El Propietario Eliminar Evento		
Pre-Condiciones	Debe haber algún Evento ingresado Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1 .El Propietario presiona la opción eliminar en el evento seleccionado.  3 .El Propietario presiona la opción aceptar.	2. El sistema pregunta si el Propietario está seguro.  4. El sistema Elimina del sistema ese Evento, solo visualmente ya que el sistema lo deja con estado anulado.	
Flujo alternativo 1	3. El Propietario presiona Cancelar.	4. El sistema cierra la interface de y vuelve a la interface principal.	
Post condiciones	El sistema elimina el evento de la vista.		

Tabla 6.7: Especificación de caso de uso Eliminar Evento

**6.1.3.6 Caso de Uso: Generar Reporte Actividades**

Caso de uso		Reporte Actividad	
ID	CU_06		
Descripción	El Propietario Genera Reporte Actividad		
Pre-Condiciones	Debe haber alguna actividad ingresada, Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario presiona la opción Reporte Actividad  2 El Propietario Ingresa los datos de los filtros que estime conveniente y presiona buscar.	2. El sistema muestra la interface de filtros de reporte.  4. El sistema realiza la consulta y visualiza todas las Actividad correspondiente según el filtro.	
Flujo alternativo 1	3. El Propietario presiona limpiar.	4. El sistema limpia el formulario y la tabla.	
Post condiciones	El sistema enlista las actividades filtradas.		

Tabla 6.8: Especificación de caso de uso Reporte Actividad

**6.1.3.7 Caso de Uso: Anular Categoría**

Caso de uso		Eliminar Categoría	
ID	CU_07		
Descripción	El Propietario eliminar Categoría		
Pre-Condiciones	Haber iniciado sesión. Tener al menos una categoría ingresada. Categoría no asociada a ninguna actividad o evento.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario presiona la opción categoría.	2.El sistema Muestra una lista de las categorías.	

	<p>3.El Propietario presiona Eliminar categoría a elección.</p> <p>5.El Propietario presiona Aceptar.</p>	<p>4 .El sistema verifica que la categoría no esté utilizada.</p> <p>6.El sistema elimina categoría ya no siendo visible en la vista.</p>
Flujo alternativo 1		La categoría esta en uso y manda un mensaje de error.
Post condiciones	El sistema elimina la categoría de la lista.	

Tabla 6.9 :Especificación de caso de uso Eliminar Categoría

#### 6.1.3.8 Caso de Uso: Generar Reporte Eventos

Caso de uso		Generar Reporte Eventos	
ID	CU_08		
Descripción	El Propietario genera un reporte Eventos		
Pre-Condiciones	Haber iniciado sesión. Debe haber algún eventos ingresado.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	<p>1. El Propietario presiona la opción Reporte Evento.</p> <p>3. El Propietario Ingresa los datos de los filtros que estime conveniente y presiona Buscar.</p>	<p>2.El sistema muestra la interface de filtros de reporte.</p> <p>4.El sistema realiza la consulta y visualiza todas las Actividad correspondiente según el filtro.</p>	
Flujo alternativo 1	2. El Propietario presiona limpiar.	3.El sistema limpia el formulario y la tabla.	
Post condiciones	El sistema enlista los eventos filtrados.		

Tabla 6.10: Especificación de caso de uso Generar Reporte Eventos

**6.1.3.9 Caso de Uso: Editar Evento**

Caso de uso		Editar Evento	
ID	CU_09		
Descripción	El Propietario editar Evento		
Pre-Condiciones	Debe haber algún evento ingresado. Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario selecciona la opción editar de un evento a elección.  3. El Propietario edita las opciones que estime conveniente y presiona guardar.	2.El sistema muestra una interfaz para poder editar el evento.  4.El sistema procesa los datos para editar el evento en la base de datos.	
Flujo alternativo 1	El Propietario no quiere editar el evento seleccionado y selecciona cancelar.	El sistema no guarda ningún cambio y direcciona a la página inicial.	
Post condiciones	El sistema edita los cambios del evento visualmente en la vista.		

Tabla 6.11: Especificación de caso de uso Editar Evento

**6.1.3.10 Caso de Uso: Editar Actividad**

Caso de uso		Editar Actividad	
ID	CU_10		
Descripción	El Propietario edita una Actividad		
Pre-Condiciones	Debe haber alguna actividad ingresada. Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos	El actor	El sistema	

Básicos	<p>1. El Propietario selecciona la opción editar  de un actividad a elección.</p> <p>3. El Propietario edita las opciones que estime conveniente y presiona el botón guardar .</p>	<p>2.El sistema muestra los campos editables</p> <p>4 El sistema procesa los datos para editar la actividad en la base de datos.</p>
Flujo alternativo 1	<p>3. El propietario presiona el botón cancelar  para no realizar cambios.</p>	<p>4. El sistema no realiza los cambios.</p>
Post condiciones	<p>El sistema edita la actividad visualmente en la lista actividades.</p>	

Tabla 6.12: Especificación de caso de uso Editar Actividad Libre

### 6.1.3.11 Caso de Uso: Crear cuenta

Caso de uso		Crear cuenta	
ID	CU_11		
Descripción	El administrador crea un cuenta o registra		
Pre-Condiciones	No deben haber una cuanta de algún propietario.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	<p>1. El propietario hace clic en a la opción registrar en la interfaz de logeo.</p> <p>3. El administrador Ingresa los datos tales como nombre y contraseña y presiona guardar</p>	<p>2.el sistema despliega la interfaz con un formulario.</p> <p>4.El sistema registra la cuenta del nuevo propietario y direcciona a la vista de inicio de sesión.</p>	

Flujo alternativo 1	3.El administrador presiona cancelar.	4. El sistema no guarda los datos y direcciona a la vista inicio de sesión.
Post condiciones	El sistema oculta la opción de registro de la vista inicio de sesión.	

Tabla 6.13: Crear cuenta

### 6.1.3.12 Caso de Uso: Editar SubActividad

Caso de uso		Editar SubActividad	
ID	CU_12		
Descripción	El Propietario edita una SubActividad		
Pre-Condiciones	Debe haber alguna SubActividades ingresada a una Actividad. Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario selecciona el botón editar  En la SubActividad de la actividad a elección.  3.El Propietario edita los datos que estime conveniente y presiona el boton guardar  .	2.El sistema muestra los campos editables  4.El sistema procesa los datos para editar la actividad en la base de datos.	
Flujo alternativo 1	3- El propietario presiona el botón cancelar  para no realizar cambios.	4. El sistema no realiza los cambios.	
Post condiciones			

Tabla 6.14: Especificación de caso de uso Editar SubActividad

**6.1.3.13 Caso de Uso: Iniciar Sesión**

Caso de uso		Iniciar Sesión	
ID	CU_13		
Descripción	El propietario inicia una sesión		
Pre-Condiciones	No debe existir una sesión iniciada anteriormente Debe a ver un propietario registrado. Debe estar en la interfaz de login.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1 .El propietario, estando en la interfaz de inicio de sesión, ingresa su usuario y contraseña.	2.El sistema realiza el proceso de autenticación.  3. Si los datos son correctos el sistema direcciona a la página principal.	
Flujo alternativo 1		3. Si los datos son incorrectos el sistema se queda en la página de iniciar sesión, y envía un advertencia al propietario.	
Post condiciones	El sistema direcciona a la vista principal.		

**Tabla 6.15: Especificación Iniciar Sesión**

**6.1.3.14 Caso de Uso: Cerrar Sesión**

Caso de uso		Cerrar Sesión	
ID	CU_14		
Descripción	El propietario cerrar una sesión		
Pre-Condiciones	El propietario debió haber iniciado sesión anteriormente El sistema no debió estar más de 30 min sin acción.		
Actores	Propietario		
Flujo de Eventos	El actor	El sistema	

Básicos	1. El propietario está en la sesión y presiona la opción Serrar Cesión en la esquina superior izquierda.	2.El sistema responderá cerrando la sesión del propietario direccionado a la lista de eventos más próximos, para luego de unos segundos direccionar a la interfaz de inicio de sesión.
Flujo alternativo 1		
Post condiciones	El sistema finalmente direcciona a la interfaz de inicio de sesión	

Tabla 6.16: Especificación Cerrar Sesión

### 6.1.3.15 Agregar SubActividad

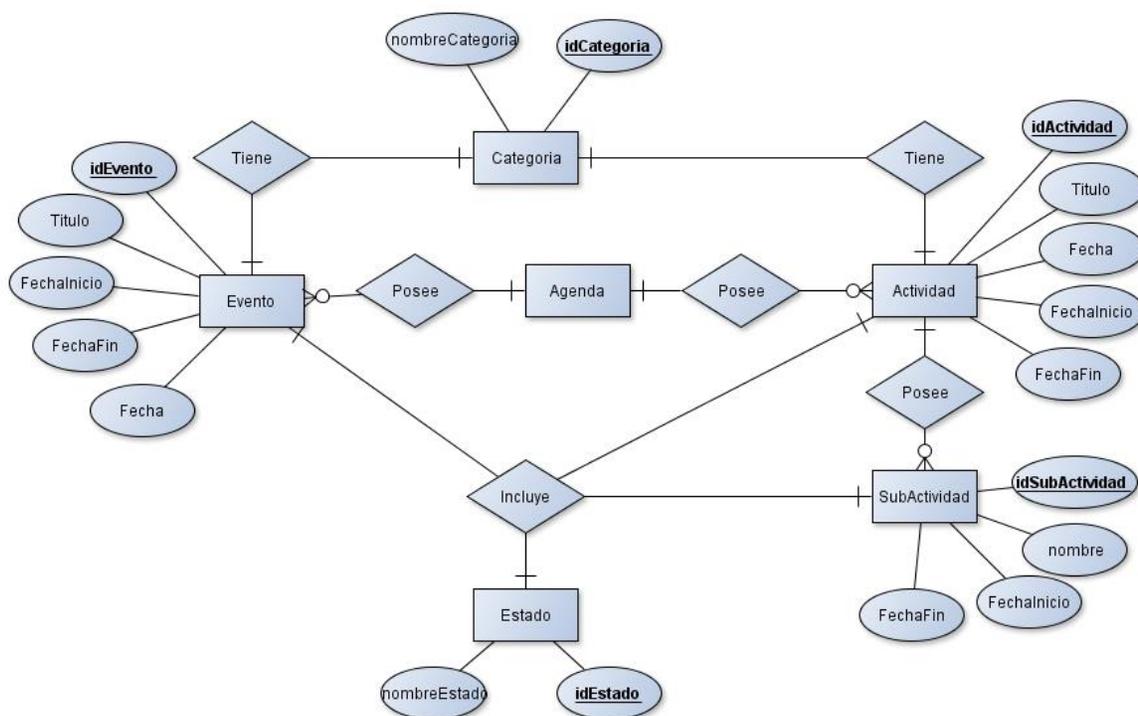
Caso de uso		Agregar SubActividad	
ID	CU_15		
Descripción	El Propietario Agrega una nueva SubActividad		
Pre-Condiciones	Debe haber alguna actividad ingresada. Haber iniciado sesión.		
Actores	Propietario		
Flujo de Eventos Básicos	El actor	El sistema	
	1. El Propietario busca la Actividad a la que desea agregar una nueva SubActividad y presiona el botón agregar .  3. El propietario escribe el titulo de la nueva SubActividad y presiona el botón agregar .	2.El sistema muestra una interfaz emergente donde se agrega las SubActividades.  4. El sistema agrega a la lista de subActividades.	

	5. El propietario luego de ingresar las SubActividades presiona el botón Guardar.	6. El sistema procesa los cambio y guarda los nuevos datos.
Flujo alternativo 1	3. El propietario sierra la ventana para no realizar cambios.	4. El sistema cierra la ventana y vuelve a la interfaz principal.
Post condiciones	El sistema agrega la subActividad a la actividad seleccionada.	

## 6.2 Modelamiento de datos

Para el Proyecto se utilizó un modelo de entidad relación para tener una percepción de como se construirá la base de datos.

En la figura 6.2 se presenta el modelo entidad relación, que indica las entidades que forman el sistema.



Figuras 6.2:Modelo Entidad Relación

### 6.2.1 Descripción de Entidades

- **Evento:**  
Entidad la cual registra los eventos del Propietario, los cuales serán calendarizados.
- **Actividad:**  
Entidad la cual registra las actividades del Propietario, el cual posee un titulo y una fecha de ingreso.

- **SubActividad:**

Esta entidad registra las subActividades de las Actividades.

- **Estado:**

Entidad la cual registra los posibles estados que pueden poseer las Actividades, SubActividades y Eventos.

- **Categoría:**

Entidad que registra las distintas categorías la cual se clasifican las Actividades y eventos.

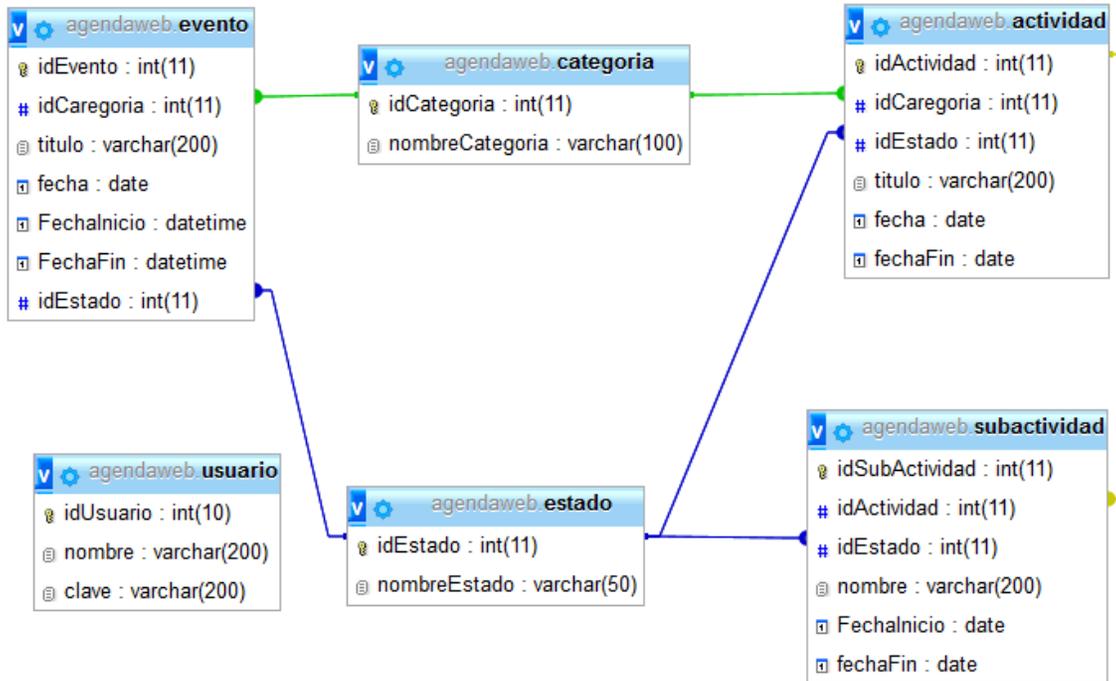
- **Agenda:**

Entidad que posee Eventos y Actividades asociadas.

## 7 DISEÑO

### 7.1 Diseño físico de la base de datos

En la figura 7.1 se aprecia el diseño físico de la base de datos, la cual se visualiza las tablas con sus respectivas relaciones, restricciones.



Figuras 7.1:Modelo Relacional de base de datos

## 7.2 Diseño interfaz y navegación

### 7.2.1 Interfaz Principal

La aplicación de agenda WEB presenta una estructura de interfaz principal sencilla que se divide en 3 áreas descritas en la figura 7.2 siguiente:



Figuras 7.2:Diseño interfaz y navegación

## 1. Área Encabezado

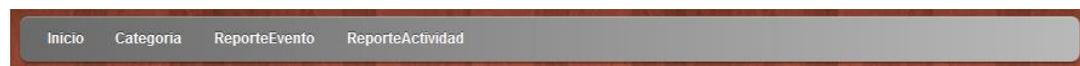
En la figura 7.3 se aprecia el área del encabezado, en la cual se encuentra la opción de cerrar sesión.



**Figuras 7.3:Área Encabezado**

## 2. Área Menú

En la figura 7.4 se aprecia el área de menú, en la cual el propietario tendrá acceso a distintas opciones generales del sistema.



**Figuras 7.4:Área Menú**

## 3. Área Contenido

La última área se divide en dos secciones, las cuales corresponden a la sección eventos y la sección actividades, descritas a continuación.

### a. Sección Eventos:

En la figura 7.5 se puede ver la sección de eventos, en la cual se visualizan todos los eventos calendarizados ingresados por el propietario, esta se puede navegar temporalmente con los botones atrás, hoy y adelante.



**Figuras 7.5:Sección Eventos**

b. Sección Actividad:

En la figura 7.6 se puede ver la sección de actividades, en la cual se visualizan las actividades ingresadas por el propietario, esta se puede navegar temporalmente con los botones atrás, hoy y adelante.



Figuras 7.6:Sección Actividad

7.2.2 Interfaz Acciones

La aplicación de agenda WEB tiene una interfaz de acción la cual posee las opciones más necesarias e imprescindibles para cumplir las necesidades de cualquier propietario, y se divide en 5 secciones.

1. Iniciar Sesión

En la figura 7.7 se muestra la interfaz de acción iniciar sesión.



Figuras 7.7:Iniciar Sesión

## 2. Crear Evento y Actividad

En la figura 7.8 se aprecia la interfaces de acción de crear un nuevo evento y Actividad, la cual pide los datos fundamentales en cada caso.

**Nuevo Evento**

**Título:**

**Inicio:**

**Fin:**

**Categoría:**

**Estado:**

**Nueva Actividad**

**Título :**

**Categoría:**

**Estado:**

**SubActividades:**

SubActividad		
Analizar temas principales	<input type="button" value="Eliminar"/>	<input type="button" value="Editar"/>
Ver unidades estudiadas	<input type="button" value="Eliminar"/>	<input type="button" value="Editar"/>

Figuras 7.8: Crear Evento y Actividad

## 3. Categoría

En la figura 7.9 se visualiza la interfaz correspondiente a las categorías, en la cual se podrán crear editar y eliminar categorías.

**Categorías**

Nombre:

1 10

Categoría		
Docencia	<input type="button" value="Eliminar"/>	<input type="button" value="Editar"/>
Ing. de Software	<input type="button" value="Eliminar"/>	<input type="button" value="Editar"/>
Investigación	<input type="button" value="Eliminar"/>	<input type="button" value="Editar"/>

1 10

Figuras 7.9: Categoría

#### 4. Reporte Evento

En la figura 7.10 se puede apreciar la interfaz de acción que genera los reportes de eventos de acuerdo a distintos filtros.

titulo	Fecha(horas)	Estado	Categoria
Prueba	17-11-2014(10:00AM-11:30AM)	Docencia	No cumplida

Figuras 7.10:Reporte Evento

#### 5. Reporte Actividad

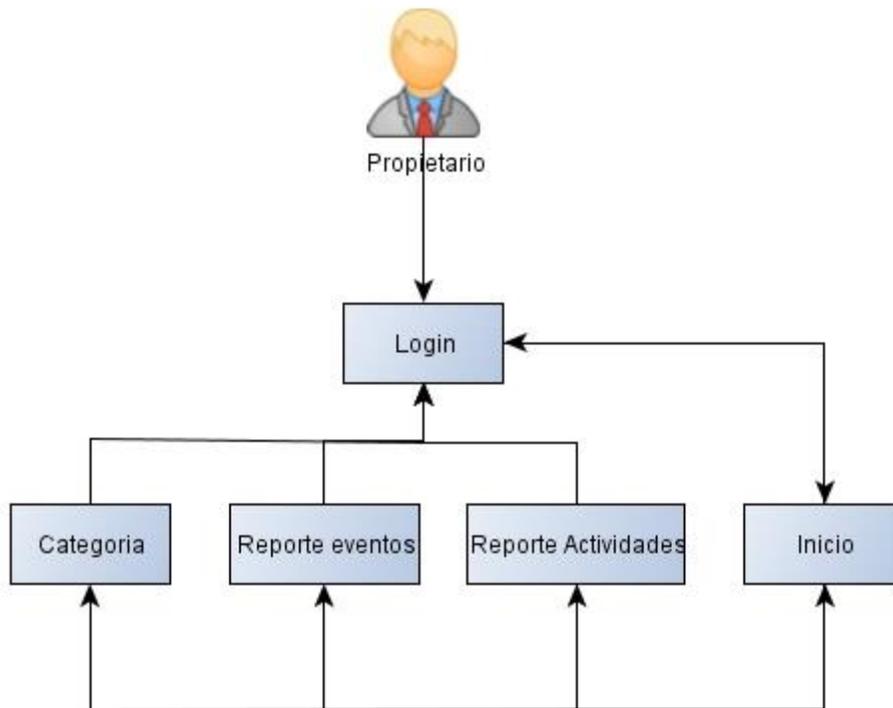
En la figura 7.11 se puede apreciar la interfaz de acción que genera los reportes de actividad de acuerdo a distintos filtros.

Actividad	Fecha	Estado	Categoria
Ing de Sw	27-10-2014	No cumplido	Ing. de Software
Ing de Sw	27-10-2014	No cumplido	Ing. de Software

Figuras 7.11:Reporte Actividad

### 7.2.3 Esquema de navegación

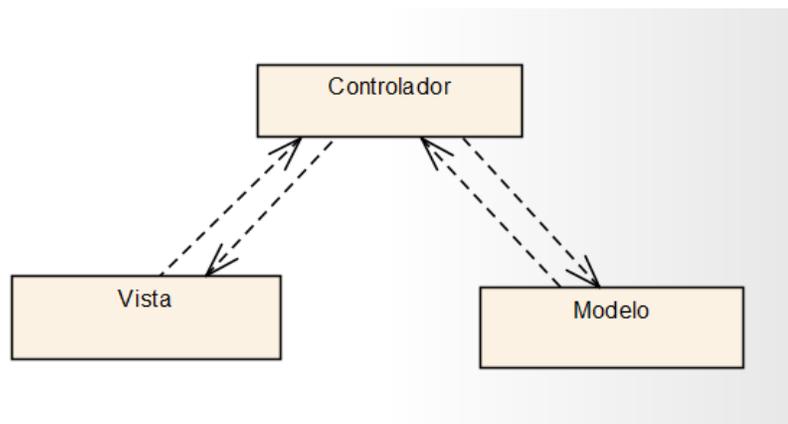
En la siguiente figura 7.12 se muestra un esquema de navegación que posee el propietario en la aplicación web, en la cual se aprecian sus principales vistas y dirección de navegación.



Figuras 7.12:Esquema de navegación

### 7.3 Diseño general de arquitectura funcional

La arquitectura general utilizada se puede visualizar en la figura 7.13 que está a continuación.



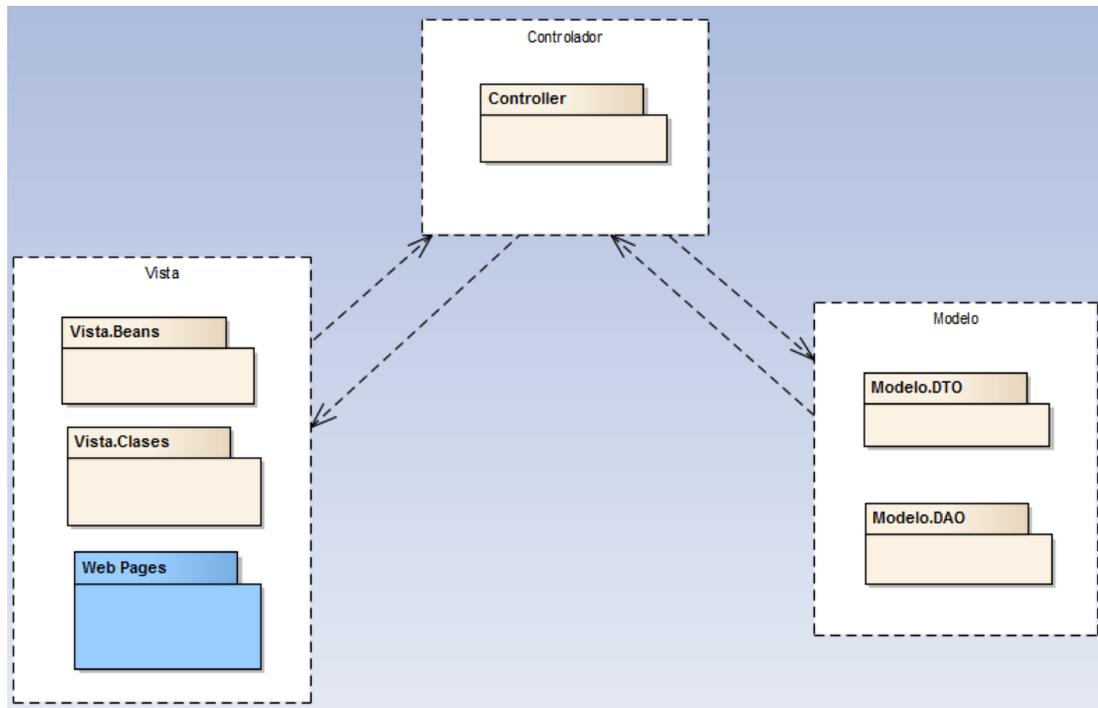
Figuras 7.13:esquema simple de arquitectura MVC

Esta arquitectura es muy buena ya que facilita el desarrollo de la aplicación, y mucho más si se trabaja en un ambiente colaborativo, la vista maneja lo referente a la interacción con el propietario del sistema, la cual genera distintas peticiones al controlador. El controlador se encarga de manejar las peticiones de la vista, ya sea procesamiento de tareas o peticiones de datos. El controlador, al recibir una petición específica, decide a quien recurrir dependiendo de lo que se solicite.

El núcleo de las funcionalidades de la aplicación se encuentra en el modelo, es la capa donde se trabaja con los datos, por lo tanto tendrá distintas acciones para acceder a la información y también para actualizar su estado, teniendo una interacción directa con la base de datos.

Estas tres capas tienen una interacción constante entre ellas, para así las funcionalidades de la aplicación funcionen perfectamente, por lo tanto, gracias a esto podemos tener rápidamente una vista ordenada y clara del sistema.

En la Figura 7.14 se muestra en una forma sencilla la estructura del sistema la cual ha sido dividida como se explicó anteriormente en las capas vista, controlador y modelo.



**Figuras 7.14:Componentes de la arquitectura del sistema**

En cada capa, diferentes paquetes que cumplen distintas funcionalidades. En la vista tenemos los beans y las clases de soporte, así como también el contenido web. En el controlador se hace la lógica de negocio entre los bean y el modelo. En la capa modelo tenemos los paquetes DTO y DAO los cuales tiene la implementación del modelo (Data Transfer Object) y el acceso de este (Data Access Object).

---

## **8 PRUEBAS:**

---

El fin de llevar a cabo las pruebas de software, es detectar los errores que pueden haber quedado en alguna etapa de diseño o construcción del proyecto, y con esto poder analizarlo y solucionarlo, para así cumplir la totalidad de objetivos del proyecto.

### **8.1 Elementos de prueba**

Los elementos sobre los cuales se ejecutarán las pruebas se dividieron en 4 partes para un mejor entendimiento y orden, y son los siguientes:

Eventos:

1. Crear Evento
2. Editar Evento
3. Eliminar Evento
4. Generar Reporte Evento

Actividad:

5. Crear Actividad
6. Editar Actividad
7. Generar Reporte Actividad
8. Editar SubActividad
9. Agregar SubActividad

Categorías:

10. Crear Categoría
11. Editar Categoría
12. Eliminar Categoría

Sesión:

13. Iniciar Sesión
14. Cerrar Sesión

## 8.2 Especificación de las pruebas

En las siguientes tablas se expondrán las especificaciones de las pruebas.

### 8.2.1 Crear Evento

<b>Caso de prueba</b>	<b>Crear Evento</b>
<b>Código de caso de prueba</b>	01
<b>Descripción de la prueba</b>	Se creará un evento nuevo
<b>Condiciones de ejecución</b>	Se debe haber iniciado sesión. Se debe estar en la interfaz principal.
<b>Entradas/pasos de ejecución</b>	1. En el sector de evento calendarizado, se presiona en el rango de hora requerido. 2. Luego que aparece el cuadro emergente con el formulario se llena con los datos requeridos(título, rango de fecha, categoría y estado) 3. se completan todos los campos y se presiona el botón Guardar.
<b>Resultado esperado</b>	2. Sistema valide rangos de fechas. 3. el sistema agrega el evento al sistema y se hace visible en el sector calendarizado de eventos si corresponde a la fecha actual. 3. Sistema valide que si no se ingrese un título no se agregue al sistema.
<b>Evaluación de la prueba</b>	-El sistema efectivamente valida rango de fechas, fecha fin no puede ser mayor a fecha inicial. -Se comprueba que un evento sin título no se ingresa al sistema. -Se ha logrado agregar el nuevo evento al sistema.
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.1: Pruebas Crear Evento

## 8.2.2 Editar Evento

<b>Caso de prueba</b>	<b>Editar Evento</b>
<b>Código de caso de prueba</b>	02
<b>Descripción de la prueba</b>	Se editará un evento cualquiera.
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>-Se debe haber iniciado sesión.</li> <li>-Se debe estar en la interfaz principal.</li> <li>-Se debe tener al menos un evento registrado en el sistema.</li> </ul>
<b>Entradas/pasos de ejecución</b>	<ol style="list-style-type: none"> <li>1. En el sector evento calendarizado, se navega cronológicamente si es necesario (atrás, hoy, adelante), luego hace clic sobre un evento a elección.</li> <li>2. Luego que aparece el cuadro emergente con el formulario, se editan los datos que se estime conveniente. (titulo, rango de fecha, categoría o estado)</li> <li>3. se completan todos los campos y se presiona el botón Guardar.</li> </ol>
<b>Resultado esperado</b>	<ol style="list-style-type: none"> <li>2. Que el sistema valide rangos de fechas.</li> <li>3. Si título se edita en blanco, no se guardan los cambios de la edición.</li> <li>3. El sistema edita los datos que ya tenía por los nuevos datos que se cambiaron.</li> </ol>
<b>Evaluación de la prueba</b>	<ul style="list-style-type: none"> <li>-El sistema efectivamente valide rango de fechas, fecha fin no puede ser mayor a fecha inicial.</li> <li>-Se comprueba que al editar el evento con un titulo en blanco no se edita en el sistema.</li> <li>-Se edito el evento exitosamente.</li> </ul>
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.2: Prueba Editar Evento

### 8.2.3 Eliminar Evento

<b>Caso de prueba</b>	<b>Eliminar Evento</b>
<b>Código de caso de prueba</b>	03
<b>Descripción de la prueba</b>	Se eliminará un evento cualquiera.
<b>Condiciones de ejecución</b>	-Se debe haber iniciado sesión. -Se debe estar en la interfaz principal. -Se debe tener al menos un evento registrado en el sistema.
<b>Entradas/pasos de ejecución</b>	1. En el sector evento calendarizado, se navega cronológicamente si es necesario (atrás, hoy, adelante), luego hace clic sobre un evento a elección.  2. Luego que aparece el cuadro emergente con el formulario se presiona el botón Eliminar  3. Se elimina el evento del sistema
<b>Resultado esperado</b>	3. El evento se elimine efectivamente del sistema y no sea visible.
<b>Evaluación de la prueba</b>	-Al seleccionar el evento aparece el cuadro emergente. -Efectivamente el evento se elimino del sistema.
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.3: Eliminar Evento

### 8.2.4 Generar Reporte Evento

<b>Caso de prueba</b>	<b>Generar Reporte Evento</b>
<b>Código de caso de prueba</b>	04
<b>Descripción de la prueba</b>	Se debe generar un reporte de eventos.
<b>Condiciones de ejecución</b>	-Se debe haber iniciado sesión. -Se debe estar en la interfaz reporte evento.

	-Se debe tener al menos un evento registrado en el sistema.
<b>Entradas/pasos de ejecución</b>	<ol style="list-style-type: none"> <li>1. Presionar la opción reporte Evento en la barra menú superior.</li> <li>2. Se visualiza la interfaz reporte evento.</li> <li>3. Se llena el formulario con distintos datos y sin datos.</li> <li>4. Se presiona el botón Buscar y se despliega la tabla con los eventos filtrados</li> <li>5. Se presiona el botón limpiar y se limpia el formulario y la tabla para una nueva búsqueda o reporte.</li> </ol>
<b>Resultado esperado</b>	<ol style="list-style-type: none"> <li>3. El rango de fechas se deben validar.</li> <li>4. El sistema al buscar con los distintos filtros muestre los datos esperados.</li> </ol> <p>Si el formulario esta en blanco y no se especifican datos, el sistema despliega la tabla con todos los eventos existentes</p> <ol style="list-style-type: none"> <li>4. Si la búsqueda no tiene resultado, se despliega mensaje informando que no hay datos en la búsqueda.</li> <li>5. Se presiona el botón Limpiar y se limpia el formulario de búsqueda y la tabla de eventos.</li> </ol>
<b>Evaluación de la prueba</b>	<p>-El sistema efectivamente valida rango de fechas, fecha fin no puede ser mayor a fecha inicial.</p> <p>-Al colocar un búsqueda que no tiene datos, se despliega mensaje alertando.</p> <p>-Se genera exitosamente el reporte evento.</p>
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.4: Prueba Generar Reporte Evento

## 8.2.5 Crear Actividad

<b>Caso de prueba</b>	<b>Crear Actividad</b>
<b>Código de caso de prueba</b>	05
<b>Descripción de la prueba</b>	Se creara una actividad nuevo
<b>Condiciones de ejecución</b>	Se debe haber iniciado sesión. Se debe estar en la interfaz principal.
<b>Entradas/pasos de ejecución</b>	<ol style="list-style-type: none"> <li>1. En el sector Actividades se presiona el botón .</li> <li>2. Luego que aparece el cuadro emergente con el formulario se llena con los datos requeridos. para una nueva actividad (título, categoría y estado)</li> <li>3. Si la nueva actividad tiene SubActividades se le pone un título y se presiona el botón , y se agrega a la lista de subactividades.</li> <li>4. Al llenar todos los datos referente a la nueva Actividad se presiona el botón guardar.</li> </ol>
<b>Resultado esperado</b>	<p>3. Se agrega una subAcatividad a la lista cada vez que colocamos un titulo y una SubActividad.</p> <p>si la Actividad no tiene un título, no se pueden agregar SubActividades.</p> <p>4.El sistema Agrega al sistema la nueva Actividad, siendo visible en el sector de actividades en la interfaz principal.</p> <p>Si la nueva Actividad no tiene un título, la Actividad no se agrega al sistema</p>
<b>Evaluación de la prueba</b>	Se ha logrado agregar la nueva Actividad al sistema.
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.5:Prueba Crear Actividad

## 8.2.6 Editar Actividad

Caso de prueba	Editar Actividad
Código de caso de prueba	06
Descripción de la prueba	Se editará una actividad cualquiera.
Condiciones de ejecución	<ul style="list-style-type: none"> <li>-Se debe haber iniciado sesión.</li> <li>-Se debe estar en la interfaz principal.</li> <li>-Se debe tener al menos una actividad registrada en el sistema.</li> </ul>
Entradas/pasos de ejecución	<ol style="list-style-type: none"> <li>1. En el sector de Actividades en la interfaz principal, se navega cronológicamente si es necesario (atrás, hoy, adelante), luego se hace clic sobre el botón Editar Actividad (  ), de la Actividad a elección que está en la tabla.</li>   <li>2. Luego la fila correspondiente a la actividad seleccionada se hacen editables sus datos (titulo y estado)</li> <li>3. Se modifica los datos estimados y se presiona el botón Editar Actividad  .</li> </ol>
Resultado esperado	<ol style="list-style-type: none"> <li>2. Que los datos al presionar la opción editar se puedan modificar.</li> <li>3. Si editamos el titulo a vacío el sistema alerta que no se puede.</li> <li>3. Al presionar el botón Editar Actividad  , los datos se actualizan en el sistema y base de datos. Si se arrepiente de la edición se presiona cancelar  , y los datos no se modifican.</li> </ol>
Evaluación de la prueba	Se editó la Actividad exitosamente.
RESULTADO	<b>Aprobado</b>

Tabla 8.6: Editar Actividad

### 8.2.7 Generar Reporte Actividad

<b>Caso de prueba</b>	<b>Generar Reporte Actividad</b>
<b>Código de caso de prueba</b>	07
<b>Descripción de la prueba</b>	Se debe generar un reporte de Actividades.
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>-Se debe haber iniciado sesión.</li> <li>-Se debe estar en la interfaz reporte Actividades.</li> <li>-Se debe tener al menos una actividad registrada en el sistema.</li> </ul>
<b>Entradas/pasos de ejecución</b>	<ol style="list-style-type: none"> <li>1. Presionar la opción ReporteActividades en la barra menú superior.</li> <li>2. Se visualiza la interfaz reporte actividades.</li> <li>3. Se llena el formulario con distintos datos y sin datos.</li> <li>4. Se presiona el botón Buscar y se despliega la tabla con las actividades filtradas.</li> <li>5. Se presiona el botón limpiar y se limpia el formulario y la tabla para una nueva búsqueda o reporte.</li> </ol>
<b>Resultado esperado</b>	<ol style="list-style-type: none"> <li>4. El sistema al buscar con los distintos filtros muestre los datos esperados.</li> </ol> <p>Si el formulario esta en blanco y no se especifican datos, el sistema despliega la tabla con todos las Actividades existentes</p> <ol style="list-style-type: none"> <li>4. Si se puede escoger si las subActividades son visible o no presionando el botón.</li> <li>5. Se presiona el botón Limpiar y se limpia el formulario de búsqueda y la tabla de Actividades.</li> </ol>
<b>Evaluación de la prueba</b>	<ul style="list-style-type: none"> <li>-Si presionamos el botón Subcategorías efectivamente puede ser seleccionado si las subActividades son visibles o no.</li> <li>-Se genera exitosamente el reporte de Actividades.</li> </ul>
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.7:Prueba Generar Reporte Actividad

### 8.2.8 Editar SubActividad

Caso de prueba	Editar SubActividad
Código de caso de prueba	08
Descripción de la prueba	Se editará una SubActividad cualquiera.
Condiciones de ejecución	<p>-Se debe haber iniciado sesión.</p> <p>-Se debe estar en la interfaz principal.</p> <p>-Se debe tener al menos una SubActividad registrado en una Actividad en el sistema.</p>
Entradas/pasos de ejecución	<p>1. En el sector de Actividades en la interfaz principal, se navega cronológicamente si es necesario (atrás, hoy, adelante), luego se hace clic sobre el botón  para desplegar las SubActividades de la Actividad seleccionada.</p> <p>2. Se edita a elección la SubActividad presionando editar SubActividad (  ), sobre la subActividad en la lista desplegada de la Actividad.</p> <p>3. Luego la fila correspondiente a la SubActividad seleccionada se hacen editables sus datos (titulo y estado)</p> <p>4. Se modifican los datos estimados y se presiona el botón Editar SubActividad  .</p>
Resultado esperado	<p>2. Los datos al presionar la opción editar se puedan modificar.</p> <p>3. Al presionar el botón Editar SubActividad  , los datos se actualizan en el sistema y base de datos.</p> <p>Si se arrepiente de la edición se presiona cancelar  , y los datos no se modifican.</p>
Evaluación de la prueba	-Efectivamente los datos se vuelven editables.

	<p>-Si se presiona  no se realiza ningún cambio.</p> <p>-Se edito la SubActividad de la Actividad exitosamente.</p>
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.8: Prueba Editar SubActividad

### 8.2.9 Agregar SubActividad

<b>Caso de prueba</b>	<b>Agregar SubActividad</b>
<b>Código de caso de prueba</b>	09
<b>Descripción de la prueba</b>	Se agregara una SubActividad a una Actividad.
<b>Condiciones de ejecución</b>	<p>-Se debe haber iniciado sesión.</p> <p>-Se debe estar en la interfaz principal.</p> <p>-Se debe tener al menos una Actividad registrado.</p>
<b>Entradas/pasos de ejecución</b>	<p>1. En el sector de Actividades en la interfaz principal, se navega cronológicamente si es necesario(atrás, hoy, adelante), luego se hace clic sobre el botón  para desplegar una interfaz con un formulario y la lista de Subactividades Actuales de la Actividad seleccionada.</p> <p>2. Se ingresa un titulo a la nueva SubActividad y se presiona el botón  para agregar a la lista.</p> <p>3. Luego de agregar las Subactividades estimadas por el propietario se presiona el botón guardar.</p> <p>4. El sistema agrega las nuevas SubActividades a la actividad.</p>
<b>Resultado esperado</b>	<p>2. Al apretar el botón agregar  la SubActividad sea visible en la lista.</p> <p>3.4 El sistema agregue las nuevas SubActividades a la Actividad</p>

<b>Evaluación de la prueba</b>	-Efectivamente las nuevas SubActividades se agregan a la Actividad, siendo visibles por el sistema en la lista.
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.9:Agregar SubActividad

### 8.2.10 Crear Categoría

<b>Caso de prueba</b>	<b>Crear Categoría</b>
<b>Código de caso de prueba</b>	10
<b>Descripción de la prueba</b>	Se creará una categoría nueva
<b>Condiciones de ejecución</b>	-Se debe haber iniciado sesión. -Se debe estar en la interfaz Categoría.
<b>Entradas/pasos de ejecución</b>	1. Presionar la opción Categoría en la barra menú superior. 2. Se visualiza la interfaz Categoría. 3. Se escribe el nombre de la nueva categoría en el formulario y se presiona el botón Agregar.
<b>Resultado esperado</b>	2. se debe visualizar todas las categorías existentes en el sistema. 3. Si la categoría no tiene título no se puede agregar una nueva categoría. 3. se agrega al sistema la nueva categoría y es visualizada en la lista de categorías,
<b>Evaluación de la prueba</b>	-Al ingresar categoría sin nombre, el sistema envía un mensaje de error. -Se ha logrado agregar la nueva categoría al sistema.
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.10:Prueba Crear Categoría

### 8.2.11 Editar Categoría

Caso de prueba	Editar Categoría
Código de caso de prueba	11
Descripción de la prueba	Se editará una Categoría cualquiera.
Condiciones de ejecución	<ul style="list-style-type: none"> <li>-Se debe haber iniciado sesión.</li> <li>-Se debe estar en la interfaz Categoría.</li> <li>-Se debe tener al menos una Categoría registrada en el sistema.</li> </ul>
Entradas/pasos de ejecución	<ol style="list-style-type: none"> <li>1. En el sector de Categoría en la interfaz Categoría se hace clic sobre el botón Editar Categoría (  ), de la Categoría a elección que está en la tabla.</li> <li>2. Luego la fila correspondiente a la Categoría seleccionada se hacen editables sus datos (nombre).</li> <li>3. Se modifica los datos estimados y se presiona el botón Editar Categoría .</li> </ol>
Resultado esperado	<ol style="list-style-type: none"> <li>2. Que al presionar la opción editar los datos de la categoría se puedan modificar.</li> <li>3. Al presionar el botón Editar Categoría , los datos se actualizan en el sistema y base de datos. Si se arrepiente de la edición se presiona cancelar , y los datos no se modifican.</li> </ol>
Evaluación de la prueba	Se editó la Categoría exitosamente.
RESULTADO	<b>Aprobado</b>

Tabla 8.11: Prueba Editar Categoría

### 8.2.12 Eliminar Categoría

<b>Caso de prueba</b>	<b>Eliminar Categoría</b>
<b>Código de caso de prueba</b>	12
<b>Descripción de la prueba</b>	-Se Elimina una Categoría cualquiera.
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>-Se debe haber iniciado sesión.</li> <li>-Se debe estar en la interfaz Categoría.</li> <li>-Se debe tener al menos una Categoría registrado en el sistema.</li> <li>-Al menos una categoría que no esté referenciada por ningún evento o actividad.</li> </ul>
<b>Entradas/pasos de ejecución</b>	<ol style="list-style-type: none"> <li>1. En el sector de Categoría en la interfaz Categoría, se hace clic sobre el botón Eliminar Categoría (  ), de la Categoría a elección que está en la tabla.</li> <li>2. La categoría es eliminada del sistema, y si no sale un mensaje de error que la Categoría está siendo utilizada.</li> </ol>
<b>Resultado esperado</b>	<ol style="list-style-type: none"> <li>2. La categoría es eliminada del sistema, ya no siendo visible en la tabla de categorías.</li> </ol> <p>Si la Categoría esta sujeta a una actividad o evento del sistema muestra un mensaje de error.</p>
<b>Evaluación de la prueba</b>	Se eliminó la Categoría exitosamente.
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.12: Prueba Eliminar Categoría

### 8.2.13 Iniciar Sesión

<b>Caso de prueba</b>	<b>Iniciar Sesión</b>
<b>Código de caso de prueba</b>	13
<b>Descripción de la prueba</b>	Se tratará de acceder al sistema
<b>Condiciones de ejecución</b>	El propietario no debe estar con una sesión activa en el sistema.
<b>Entradas/pasos de ejecución</b>	<ol style="list-style-type: none"> <li>1. Se ingresa a la página de inicio de sesión.</li> <li>2. Se ingresa un Usuario y un Contraseña que no están registrados en la base de datos.</li> <li>3. Se ingresa con un Usuario y un Contraseña inválido.</li> <li>4. Se ingresa con un Usuario y un Contraseña válidos.</li> </ol>
<b>Resultado esperado</b>	<ul style="list-style-type: none"> <li>-En el paso 2 y 3 el sistema debe enviar un mensaje de advertencia indicando que el usuario o la contraseña son incorrectos.</li> <li>-En el paso 4 el sistema debe autenticar al usuario y enviarlo a la página de inicio.</li> </ul>
<b>Evaluación de la prueba</b>	<ul style="list-style-type: none"> <li>-Al ingresar un usuario y clave inválido el sistema arroja un mensaje de error.</li> <li>-Al ingresar un usuario y clave válido el sistema autentifica y direcciona a la página inicial del sistema.</li> </ul>
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.13: Prueba Iniciar Sesión

### 8.2.14 Cerrar Sesión

<b>Caso de prueba</b>	<b>Cerrar Sesión</b>
<b>Código de caso de prueba</b>	14
<b>Descripción de la prueba</b>	Se cerrará la sesión de usuario y eliminar sesión activa.
<b>Condiciones de ejecución</b>	El propietario que ejecuta la prueba debe tener una sesión iniciada en el sistema
<b>Entradas/pasos de ejecución</b>	1. se presiona la opción de la esquina superior izquierda que dice Cerrar Sesión.
<b>Resultado esperado</b>	1. La sesión es finalizada y el usuario es direccionado a una interfaz donde muestra la lista de los eventos más próximos, para luego de unos segundos direccionar a la página de inicio de sesión.
<b>Evaluación de la prueba</b>	Se cierra la sesión correctamente
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.14:Prueba Cerrar Sesión

### 8.2.15 Compatibilidad web

<b>Caso de prueba</b>	<b>Compatibilidad WEB</b>
<b>Código de caso de prueba</b>	15
<b>Descripción de la prueba</b>	Se probará compatibilidad en los navegadores más comunes. -Google Chrome -Mozilla Firefox -Internet Explorer
<b>Condiciones de ejecución</b>	-Tener conexión a internet. -Tener disponible en la computadora los 3

	navegadores.
<b>Entradas/pasos de ejecución</b>	1. Se navegará la aplicación generalmente y se verá su compatibilidad en los tres navegadores.
<b>Resultado esperado</b>	Que en los tres navegadores no exista mayor diferencia o incompatibilidad.
<b>Evaluación de la prueba</b>	-Google Chrome: Se ejecuta con éxito (nivel Bueno). -Mozilla Firefox: Se ejecuta con éxito un pequeñas variantes graficas (nivel bueno) -Internet Explorer: Se pueden ejecutar todas las funcionalidades, pero visualmente la barra menú superior se desajusta a su posición (nivel bajo)
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.15:Compatibilidad WEB

### 8.2.16 Compatibilidad Responsive

<b>Caso de prueba</b>	<b>Compatibilidad Responsive</b>
<b>Código de caso de prueba</b>	16
<b>Descripción de la prueba</b>	Se probará compatibilidad en 3 dispositivos:  -Celular (Android 4.3 ,Samsung Galaxy s3) -Tablet (Android 4.4.4 ,Asus nexus 7) -Computador (windows 7 ,Notebook Samsung rf411)
<b>Condiciones de ejecución</b>	-Tener conexión a internet. -Tener disponible los 3 dispositivos. -Tener al menos 1 navegador en cada dispositivo.

<b>Entradas/pasos de ejecución</b>	1. En cada dispositivo se navegara en la aplicación generalmente y se verá su compatibilidad en los tres dispositivos.
<b>Resultado esperado</b>	<ul style="list-style-type: none"> <li>-Que en los tres dispositivos no exista mayores incompatibilidades.</li> <li>-Vista y resolución se adapte a todos los dispositivos.</li> </ul>
<b>Evaluación de la prueba</b>	<p>Celular:</p> <ul style="list-style-type: none"> <li>-La resolución general se adapta con éxito.</li> <li>-Las tablas al tener mucha información se ve un poco comprimida, pero entendible.</li> <li>-Se pueden realizar todas las acciones</li> </ul> <p>Tablet:</p> <ul style="list-style-type: none"> <li>-La resolución general se adapta con éxito.</li> <li>-Se pueden realizar todas las acciones</li> </ul> <p>Computador:</p> <p>Todas las acciones con éxito.</p>
<b>RESULTADO</b>	<b>Aprobado</b>

Tabla 8.16:Compatibilidad Responsive

### **8.3 Responsables de las pruebas**

La responsabilidad de la ejecución de las distintas pruebas realizadas, estuvieron a cargo de los desarrolladores del proyecto.

- César Guzmán Cádiz

### **8.4 Conclusiones de pruebas**

En el transcurso de todas las múltiples pruebas surgieron problemas los cuales fueron solucionadas en el momento, dando a entender que esta etapa del desarrollo es muy importante, ya que se pueden desentrañar problemas que se quedaron y no habían sido encontrados a simple vista.

Finalmente como resultado se encontraron y arreglaron todos los problemas encontrados, Tales como formatos y validación de fechas, validar que datos sin título no se puedan agregar, errores de consultas base de datos, refrescos y visualización de datos, agregar mensajes de errores o advertencia para mejor el entendimiento de este, mejorar compatibilidad responsive, etc . Por lo tanto cada pruebas se realizo hasta tener resultados exitosos, es decir la aplicación luego de todo este proceso cumple con los objetivos del proyecto.

---

## 9 FACILIDAD DE USO

---

Para evaluar la facilidad de uso se han definido 5 etapas, basadas en el criterio de prueba de facilidad de uso en la aplicación de la ingeniería web (PRESSMAN, 2006), las cuales tendrán como resultado un mecanismo que ayudara a comprender el nivel de facilidad de uso de la aplicación web.

### 9.1 Definición de etapas evaluación

#### 9.1.1 Etapa 1

En la primera etapa se definirán un conjunto de categorías de facilidad de uso y se identificarán las metas para cada una, estas se enumerarán ya que serán utilizadas en el cuestionario que se aplicara a los distintos participantes, siendo cada una de estas evaluadas según su grado de conformidad.

##### **Interactividad:**

1. Al usar la aplicación ¿Los mecanismos de interacción son fáciles de entender o usar?( por ejemplo menús desplegados, botones, punteros, textos )

##### **Plantilla:**

2. ¿Los mecanismos de navegación, contenido y funciones están colocados en una forma que le permite a usted encontrarlo rápidamente?

##### **Legibilidad:**

3. Usted considera que, ¿El texto está bien escrito y es comprensible?
4. Usted considera que, ¿La presentación gráfica es fácil de entender?

##### **Estética**

5. Usted considera que ¿la plantilla, el color, las características relacionadas conducen a un uso más sencillo?
6. Usted ¿se siente cómodo con la apariencia y la percepción de la aplicación web?

##### **Características de despliegue**

7. Usted considera que, ¿la aplicación web utiliza en forma óptima el tamaño y la resolución de la pantalla?

##### **Sensibilidad de tiempo**

8. Usted considera que, ¿Las características, funciones y contenido importante pueden utilizarse o adquirirse de manera oportuna?

## **Aprendizaje**

9. Luego de realizar las pruebas considera que, ¿Es fácil de aprender las distintas funciones de la aplicación web?

### **9.1.2 Etapa 2**

En esta etapa se define el grado de abstracción de la prueba de facilidad de uso, ya que se puede llevar a cabo en distintos grados, ya sea en una interfaz específica (por ejemplo, un formulario), una página completa (Mecanismos y funciones de dicha interfaz) o en la aplicación web en general, siendo esta última la seleccionada como grado de abstracción, debido a que la aplicación no es tan compleja.

### **9.1.3 Etapa 3**

En esta etapa se seleccionan los participantes que ejecutarán las pruebas y el cuestionario. Los individuos que ejecutaran estas pruebas serán escogidos arbitrariamente, y se da por hecho que tienen conocimiento básico en computación y navegación web.

El número de participantes puede variar entre 5 a 20 personas.

### **9.1.4 Etapa 4**

En esta etapa se instrumentará la interacción de los participantes con la aplicación web. Los participantes realizarán una serie de pruebas programadas para que así tengan un conocimiento general del uso de las funcionalidades de la aplicación.

#### **Pruebas:**

#### **1. Iniciar sesión en aplicación web e ingresar a la aplicación**

Ingresando Usuario: Admin y Contraseña:123, y finalmente presionar el botón Iniciar.

Si los datos están bien escritos el sistema direccionará a la vista principal, y sino informará la causa del problema.

#### **2. Ingresar evento**

En la vista principal en el sector evento ingresar los siguientes eventos:

##### **a) Evento 1:**

Ingrese un evento del día de hoy con los siguientes datos:

Hacer clic sobre el sector evento calendarizado preferentemente sobre la hora en la cual se realizará el evento, con los siguientes datos:

- Títulos: Estudio grupal
- Fecha inicio: fecha de hoy a las 12:00 pm.
- Fecha fin: fecha de hoy a las 2:00 pm.

- Categoría: Estudio
- Estado: no cumplida

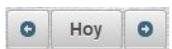
Luego de ingresar todos los datos del nuevo evento, presione el botón guardar, este podrá ser visualizado inmediatamente en el horario que corresponde.

**b) evento 2:**

Hacer clic sobre el sector de eventos e ingrese un evento, el cual se realizará el día posterior a hoy, con los siguientes datos:

- Títulos: Certamen 1
- Fecha inicio: fecha de mañana a las 10:00 am
- Fecha fin: fecha de mañana a las 11:00 am
- Categoría: Cálculo
- Estado: no cumplida

Finalmente presione el botón guardar, este será visible si busca cronológicamente



**3. Ingresar Actividad**

Ir al sector de Actividades y presionar el botón hoy  para verificar que nos encontramos cronológicamente en el día actual, luego hacer clic en el botón , e ingresar los siguientes datos

- Títulos: Estudiar
- Categoría: Cálculo
- Estado: no cumplida
- SubActividad: (Para agregar una subActividad anotar su título y presionar el botón , y será visible en la tabla)
  - Buscar libros
  - Leer apuntes

Finalmente, luego de agregar los datos y subActividades de la nueva Actividad, presionar el botón guardar, siendo visible en el sector actividades.

#### 4. Cambiar a estado cumplido una Actividad

En el sector Actividades:

- Buscar Actividad llamada : Encuesta de usabilidad
- presionar el botón editar  y cambiamos el estado a cumplida.
- Finalmente presionar el botón aceptar , y el sistema guardará los cambios.

Luego, comprobar navegando cronológicamente que mañana la actividad ya no será visible, ya que fue cumplida.

##### 9.1.5 Etapa 5

En esta etapa se define el mecanismo para valorar la facilidad de uso, luego de una investigación y encontrar un mecanismo claro y acorde a las necesidades de esta evaluación de facilidad de uso, se toma como referencia la metodología de medición y evaluación de la usabilidad en sitios web educativos (Obeso, 2005).

Se evaluará la facilidad de uso valorando las categorías y metas definidas en la Etapa 1, esto por medio de una valorización numérica entre 1 a 5, que el participante estimara según su experiencia al ejecutar las pruebas definidas en la etapa 4.

En la tabla 9.1 muestra la interpretación de cada rango.

Rango	Significado de valor
1	Fuertemente en desacuerdo
2	Algo en desacuerdo
3	Indeciso
4	Algo de acuerdo
5	Fuertemente de acuerdo

**Tabla 9.1: Interpretación de cada rango**

El valor asignado por los participantes debe ser normalizado, y se convierte en una *Puntuación Elemental (pe)* sobre una escala de 0 a 100. Este indica el grado de conformidad de facilidad de uso con respecto a las categorías y metas de la aplicación web evaluada (es decir, la puntuación elemental no está referida al valor asignado directamente por el usuario sino al grado de cumplimiento de facilidad de uso establecido) y es interpretado como un porcentaje de conformidad:

Rango	Puntuación elemental(%)
1	0
2	25
3	50
4	75
5	100

$$pe = \begin{cases} 0\% & \text{Inconformidad total} \\ 0 \leq pe \leq 100\% & \text{Conformidad parcial} \\ 100\% & \text{Completa conformidad} \end{cases}$$

**Tabla 9.2:Puntuación elemental(%)**

En los criterios de aceptabilidad hay tres regiones en un rango de puntuación de 0 a 100%, los cuales son nivel pobre, nivel aceptable y nivel bueno, como se muestran en la figura 9.1.

$$\text{Puntuación global del sitio} \begin{cases} pg < 40, & \text{nivel pobre} \\ 60 \leq pg \leq 40, & \text{nivel aceptable} \\ > 60, & \text{nivel bueno} \end{cases}$$

**Figuras 9.1:criterios de aceptabilidad**

Ya normalizando y teniendo la puntuación elemental de cada evaluación correspondiente a cada meta, ya se pueden realizar los análisis:

**Análisis 1:**

Primero se saca el PE total por categoría de cada uno de los participantes, mediante el promedio de las puntuaciones de las metas de cada categoría( $pe_1, pe_2, \dots, pe_n$ ), es decir:

$$PE = \frac{\sum_{i=1}^n pe_i}{n}$$

**Ecuación 9.1**

Donde:

$pe_i$ = Puntuaciones elementales por cada meta valorada por el participante ya normalizada, es decir, el grado de conformidad porcentual de facilidad de uso de cada meta.

$n$ = Número de metas que tiene cada categoría.

$PE$ = Puntuación elemental total promedio de cada categoría, es decir, porcentaje de conformidad de cada categoría estimada por cada participante.

**Análisis 2:**

Luego de obtener el promedio de cada categoría por participante, se saca el PE total de la aplicación web por categoría ( $PE_{total\ categoría}$ ), mediante el promedio de las puntuaciones de PE de cada categoría ( $PE_1, PE_2, \dots, PE_n$ ) asignada por los participantes, las cuales se calcularon anteriormente, es decir:

$$PE_{total\ categoría} = \frac{\sum_{i=1}^n PE_i}{n}$$

**Ecuación 9.2**

Donde:

$PE_{total\ categoría}$ = Puntuación elemental total por categoría, es decir, el grado de conformidad porcentual total de facilidad de uso por categoría de la aplicación web.

$PE_i$ = Puntuaciones elementales promedio de cada participante por categoría, es decir, el grado porcentual de conformidad de facilidad de uso por categoría asignada por cada participante.

$n$ =Número de participantes.

**Análisis 3:**

Después se calcula el  $PE_{total}$  por participante, mediante el promedio total de las puntuaciones de las categorías sacados anteriormente ( $PE_{total\ categoría\ 1}, PE_{total\ categoría\ 2}, \dots, PE_{total\ categoría\ n}$ ) por cada participante, es decir, el grado porcentual de conformidad total de facilidad de uso de la aplicación por participante, esto representado como:

$$PE_{total\ participante} = \frac{\sum_{i=1}^n PE_{total\ categoría\ i}}{n}$$

**Ecuación 9.3**

Donde:

$PE_{total\ participante}$  = Puntuación elemental total promedio por participante, es decir, el grado de conformidad porcentual total de facilidad de uso por participante de la aplicación web.

$PE_{total\ categoría\ n}$  = Puntuaciones elementales promedio de cada categoría, es decir, el grado de conformidad porcentual total de facilidad de uso por categoría de la aplicación web.  
 $n$  = Número de categorías.

#### **Análisis 4:**

Finalmente se calcula  $PE_{Global}$  de la aplicación web, mediante el promedio de puntuación elemental total por participante ( $PE_{total\ participante\ 1}, PE_{total\ participante\ 2}, \dots, PE_{total\ participante\ n}$ ), es decir, el grado porcentual de conformidad global de facilidad de uso de la aplicación web, esto representado como:

$$PE_{Global} = \frac{\sum_{i=1}^n PE_{total\ participante\ i}}{n}$$

**Ecuación 9.4**

Donde:

$PE_{Global}$  = Puntuación elemental global de la aplicación, es decir, el grado de conformidad porcentual global de facilidad de uso de la aplicación web.

$PE_{total\ participante\ i}$  = Puntuaciones elementales promedio de cada participante, es decir, el grado de conformidad porcentual total de facilidad de uso de la aplicación web por participante.

$n$  = Número de participantes.

## 9.2 Resultados

Luego de realizar el cuestionario(anexo 2 "Cuestionario de facilidad de Uso") a 10 participantes, los cuales tenían los conocimientos necesarios en computación y navegación web, además de pertenecer a variadas áreas de estudio, tales como humanista, ciencia, arte e ingeniería. Este cuestionario no se extendía a más de 10 min por persona, y luego finalmente se realizó un análisis con los múltiples datos obtenidos.

Para una mejor comprensión del análisis que se realizara a continuación, se asignó un color a cada categoría, este será utilizado según el significante de los datos de cada categoría.

A continuación se expone en la tabla 9.3 los resultados obtenidos luego de realizar el cuestionario a los 10 participantes, los cuales evaluaron el grado de conformidad de cada meta con un valor numérico del 1 al 5, explicado anteriormente en la Etapa 5 del análisis.

Participante	CATEGORIAS								
	Interactividad	Plantilla	Legibilidad		Estética		Características de despliegue	Sensibilidad de tiempo	Aprendizaje
	$MT_1$	$MT_2$	$MT_3$	$MT_4$	$MT_5$	$MT_6$	$MT_7$	$MT_8$	$MT_9$
Nº 1	4	5	5	4	5	5	5	5	4
Nº 2	5	5	5	5	5	5	5	5	5
Nº 3	4	5	5	5	4	5	4	5	5
Nº 4	5	5	5	5	5	5	5	5	5
Nº 5	5	5	5	5	5	4	5	5	5
Nº 6	5	5	4	5	5	5	5	5	5
Nº 7	5	5	5	4	5	5	5	5	5
Nº 8	5	5	5	5	4	5	4	5	5
Nº 9	5	5	5	5	5	4	5	5	5
Nº 10	5	5	4	5	5	5	4	5	5

**Tabla 9.3:Resultados de los cuestionarios**

$MT_n$ : Metas de las categorías.

El valor asignado por los participantes debe ser normalizado, y se convierte en una *Puntuación Elemental (pe)* sobre una escala de 0 a 100, por lo tanto, la tabla 9.4 muestra los datos ya normalizados para los análisis posteriores.

Participante	CATEGORIAS								
	Interactividad	Plantilla	Legibilidad		Estética		Características de despliegue	Sensibilidad de tiempo	Aprendizaje
	$pe_1$	$pe_2$	$pe_3$	$pe_4$	$pe_5$	$pe_6$	$pe_7$	$pe_8$	$pe_9$
Nº 1	75	100	100	75	100	100	100	100	75
Nº 2	100	100	100	100	100	100	100	100	100
Nº 3	75	100	100	100	75	100	75	100	100
Nº 4	100	100	100	100	100	100	100	100	100
Nº 5	100	100	100	100	100	75	100	100	100
Nº 6	100	100	75	100	100	100	100	100	100
Nº 7	100	100	100	75	100	100	100	100	100
Nº 8	100	100	100	100	75	100	75	100	100
Nº 9	100	100	100	100	100	75	100	100	100
Nº 10	100	100	75	100	100	100	75	100	100

**Tabla 9.4: datos normalizados a puntuación elemental**

$pe_n$ : puntuaciones elementales por cada meta valorada por el participante ya normalizada.

Luego de normalizar los datos se empieza con los análisis de estos, explicado anteriormente en la Etapa 5 del análisis.

### 9.2.1 Análisis 1

En el primer análisis se calcula  $PE_{T.C.n}$  total por categoría de cada uno de los participantes, mediante el promedio de las puntuaciones de las metas por categoría ( $pe_1, pe_2, \dots, pe_n$ ), es decir, el porcentaje de conformidad de cada categoría estimada por cada participante, dando como resultado tabla 9.5.

Participante	CATEGORIAS						
	Interactividad	Plantilla	Legibilidad	Estética	Características de despliegue	Sensibilidad de tiempo	Aprendizaje
	$PE_{T.C.1}$	$PE_{T.C.2}$	$PE_{T.C.3}$	$PE_{T.C.4}$	$PE_{T.C.5}$	$PE_{T.C.6}$	$PE_{T.C.7}$
Nº 1	75	100	87.5	100	100	100	75
Nº 2	100	100	100	100	100	100	100

Nº 3	75	100	100	87.5	75	100	100
Nº 4	100	100	100	100	100	100	100
Nº 5	100	100	100	87.5	100	100	100
Nº 6	100	100	87.5	100	100	100	100
Nº 7	100	100	87.5	100	100	100	100
Nº 8	100	100	100	87.5	75	100	100
Nº 9	100	100	100	87.5	100	100	100
Nº 10	100	100	87.5	100	75	100	100

**Tabla 9.5:puntuaciones elementales de categorías por participante**

### 9.2.2 Análisis 2

En el segundo análisis se calcula PE total de la aplicación web por categoría, mediante el promedio de las puntuaciones de PE de cada categoría ( $PE_{T.C 1.1}, PE_{T.C 1.2}, \dots, PE_{T.C n.n}$ ) asignada por los participantes, las cuales se calcularon anteriormente, dando como resultado la tabla 9.6.

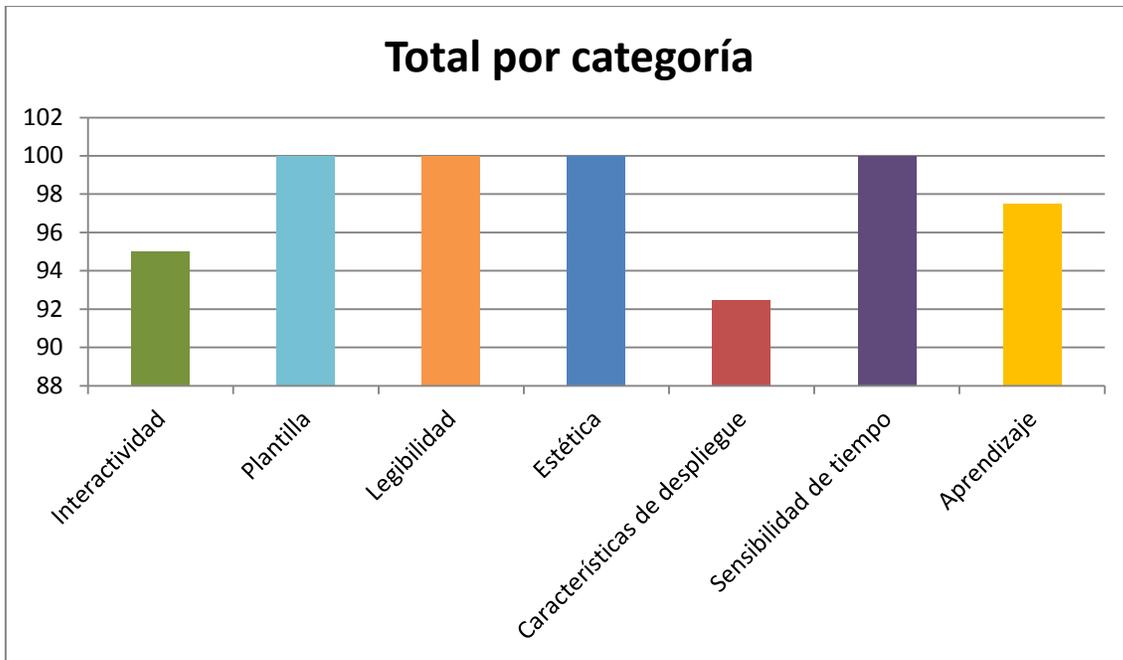
Interactividad	Plantilla	Legibilidad	Estética	Características de despliegue	Sensibilidad de tiempo	Aprendizaje
$PE_{T.C 1}$	$PE_{T.C 2}$	$PE_{T.C 3}$	$PE_{T.C 4}$	$PE_{T.C 5}$	$PE_{T.C 6}$	$PE_{T.C 7}$
95	100	100	100	92,5	100	97,5

**Tabla 9.6:puntuaciones elementales por categoría**

$PE_{T.C}$ = Puntuación elemental de categoría total

teniendo como resultado el grado de conformidad porcentual de facilidad de uso por categoría de la aplicación web.

En la figura 9.2 se muestra el grafico que muestra más claramente la comparativa de los grados de conformidad porcentual de cada categoría.



**Figuras 9.2: Grafico: Total por categoría**

### 9.2.3 Análisis 3

En el tercer análisis se calcula el  $PE_{total}$  por participante, mediante el promedio total de las puntuaciones de las categorías sacados en el primer análisis ( $PE_{T.C.1}, PE_{T.C.2}, \dots, PE_{T.C.n}$ ) por cada participante, teniendo como resultado el grado porcentual de conformidad de facilidad de uso de la aplicación por participante, en la tabla 9.7.

Categorías	$PE_{total}$
Nº 1	91,7
Nº 2	100,0
Nº 3	91,7
Nº 4	100,0
Nº 5	100,0
Nº 6	100,0
Nº 7	100,0
Nº 8.	95,8
Nº 9	100,0
Nº 10	95,8

**Tabla 9.7: puntuaciones elementales por participante**

#### 9.2.4 análisis 4

En el último análisis se calcula  $PE_{Global}$  de la aplicación web, mediante el promedio de puntuación elemental total por participante ( $PE_{total\ participante\ 1}, PE_{total\ participante\ 2}, \dots, PE_{total\ participante\ n}$ ) sacadas del análisis 3, Teniendo como resultado el grado porcentual de conformidad global de facilidad de uso de la aplicación web:

$PE_{Global} =$	97,5
-----------------	------

**Tabla 9.8: grado porcentual de conformidad global**

#### 9.2.5 Conclusión

Luego de realizar estudio de facilidad de uso se puede concluir que el promedio de facilidad de uso es aceptable y tiene un nivel bueno

---

## 10 CONCLUSION

---

Luego de terminar este sistema se puede concluir que cualquier proyecto por más pequeño o grande que sea siempre es un gran desafío, en el cual se pueden encontrar múltiples problemas de diferentes niveles de complejidad, pero si se buscan las herramientas y tecnologías adecuadas para el desarrollo, finalmente se podrá llegar a un objetivo que cumpla con las necesidades del cliente.

Durante el transcurso de este proyecto se pudo deslumbrar que la metodología de desarrollo cascada empleada no se adaptó muy bien, ya que hubo un enfoque concentrado de programación dejando para el final la documentación, además el cliente fue integrado al desarrollo del proyecto, ayudando esto a la mejor adaptación del sistema, sin embargo podían surgir múltiples nuevos requerimientos o adaptaciones de estos en cualquier etapa, finalmente se puede decir que la metodología de desarrollo fue similar a las metodologías ágiles, tales como Scrum o xp.

Personalmente lo que pude concluir es que todo proyecto es complejo, y mucho más si se realiza individualmente como en mi caso, puedo suponer claramente que cualquier proyecto que se realice en grupo y con un buen trabajo en equipo se podrían lograr mejores resultados en menos tiempo, también pude comprender que al utilizar complementos, herramientas, tecnologías y la arquitectura indicada, el desarrollo puede ser mucho más rápido y eficiente.

Este proyecto fue una gran experiencia personal, debido a que me enfrente a múltiples problemas que tuve que solucionar con mucho trabajo y paciencia, otra cosa importante que pude entender es lo primordial de escuchar al cliente, ya que cuanto más lo comprendamos el resultado final será lo más cercano a lo que el cliente se está imaginando.

---

## 11 BIBLIOGRAFÍA

---

Çivici, Ç. (2014). *USER GUIDE 5.0*.

Corporation, O. (2014). *Netbeans*. Recuperado el 2014, de Netbeans : <https://netbeans.org/>

fowler, M. *UML gota a gota*. pearson education.

notepad. (2014). *notepad*. Recuperado el 2014, de <http://notepad-plus-plus.org/>

Obeso, M. E. (2005). *Metodología de Medición y Evaluación de la Usabilidad en Sitios Web Educativos*.

Oracle. (2014). *java*. Recuperado el 2 de julio de 2014, de java: [https://www.java.com/es/download/whatis\\_java.jsp](https://www.java.com/es/download/whatis_java.jsp)

Pressman S, R. (2002). *Ingeniería de software, un enfoque práctico, 5ta Edición*. México: McGraw-Hill.

PRESSMAN, R. (2006). *Ingeniería del software: un enfoque práctico sexta edicion* (6 ed.). McGraw Hill/Interamericana Editores.

w3c. (2014). *w3c*. Recuperado el 2014, de <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>

---

## **12 ANEXO1 : MANUAL DE USUARIO**

---

---

**13 ANEXO2 : CUESTIONARIO DE FACILIDAD DE USO**

---

---

## **14 ANEXO3 : MANUAL DE RÉPLICA**

---