



UNIVERSIDAD DE QUINTANA ROO
DIVISIÓN DE CIENCIAS E INGENIERÍA

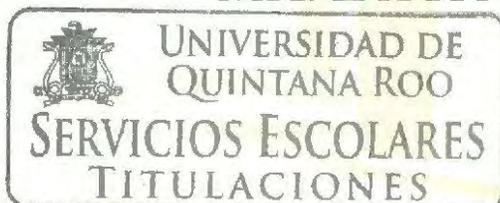
SISTEMA PARA LA GESTIÓN DE
TRABAJOS DE TITULACIÓN EN LA
DIVISIÓN DE CIENCIAS E INGENIERÍA

TESIS
PARA OBTENER EL GRADO DE
INGENIERO EN REDES

PRESENTA
FRANCISCO JAVIER ALCOCER UH

DIRECTORA DE TESIS
MTI. MELISSA BLANQUETO ESTRADA

ASESORES
MTI VLADIMIR VENIAMIN CABAÑAS VICTORIA
LIC. SERGIO ALBERTO SOLÍS SOSA
MSI. RUBÉN ENRIQUE GONZÁLEZ ELIXAVIDE
MSI. LAURA YÉSICA DÁVALOS CASTILLA



CHETUMAL QUINTANA ROO, MÉXICO, JUNIO DE 2015

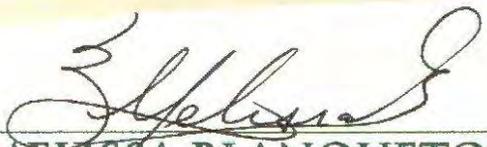


UNIVERSIDAD DE QUINTANA ROO
DIVISIÓN DE CIENCIAS E INGENIERÍA

TRABAJO DE TESIS ELABORADO BAJO SUPERVISIÓN DEL
COMITÉ DE ASESORÍA Y APROBADO COMO REQUISITO
PARCIAL PARA OBTENER EL GRADO DE:
INGENIERO EN REDES

COMITÉ DE TRABAJO DE TESIS

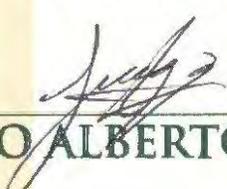
DIRECTORA:


MTI. MELISSA BLANQUETO ESTRADA

ASESOR:


MTI. VLADIMIR VENIAMIN CABAÑAS
VICTORIA

ASESOR:


LIC. SERGIO ALBERTO SOLÍS SOLÍS

UNIVERSIDAD DE QUINTANA ROO

 DIVISIÓN DE
CIENCIAS E
INGENIERÍA



UNIVERSIDAD DE
QUINTANA ROO
SERVICIOS ESCOLARES
TITULACIONES

CHETUMAL QUINTANA ROO, MÉXICO, JUNIO DE 2015

Agradecimientos

A mis padres Francisco Javier Alcocer Medina y Maribel Uh Llanes por todo su apoyo y cariño que me han brindado toda mi vida.

A mi hermana Arleny por sus consejos.

A mi directora de tesis M.T.I Melissa Blanqueto Estrada por su paciencia, enseñanzas y ayuda durante la realización de este proyecto.

A mis asesores M.T.I. Vladimir Veniamin Cabañas Victoria y Lic. Sergio Alberto Solís Sosa por su ayuda en este proyecto.

A mis amigos con quienes pasamos momentos difíciles pero también agradables.

Dedicatoria

Hoy les dedico este proyecto que con mucho esfuerzo y tiempo he logrado culminar.

Gracias por sus consejos.

Gracias por enseñarme sobre humildad y sacrificio.

Pero sobre todo gracias por estar a mi lado.

Maribel Uh Llanes

Francisco Javier Alcocer Medina

Resumen

Actualmente el uso de las tecnologías de la información y comunicación, permiten a las personas desarrollar sistemas que ayuden a automatizar y administrar procesos o tareas que se realizan cotidianamente.

En este proyecto se realiza una aplicación web para la gestión de los trabajos de titulación a nivel licenciatura de la División de Ciencias e Ingeniería.

La aplicación está desarrollada bajo el patrón arquitectónico MVC (Modelo-Vista-Controlador) por sus siglas en inglés, con ayuda del *Framework* PHP *CodeIgniter*.

La función principal de la aplicación es proporcionar una herramienta administrativa que permita registrar, administrar y proporcionar estadísticas de los trabajos de titulación a los profesores, alumnos y asistentes administrativos de la División de Ciencias e Ingeniería.

Contenido.

Agradecimientos.....	III
Dedicatoria	IV
Resumen.....	V
Contenido.....	VI
Índice de figuras.	IX
Índice de tablas.....	X
CAPÍTULO 1 INTRODUCCIÓN	1
1.1 Descripción del problema.....	2
1.2 Objetivos.....	2
Objetivo general.....	2
Objetivos específicos.....	3
1.3 Justificación.....	3
1.4 Alcance y restricciones.	3
CAPÍTULO 2 MARCO TEÓRICO.....	4
2.1 Usabilidad de una aplicación web	5
Conceptos básicos de usabilidad	5
Facilidad de manejo de la aplicación	5
Equilibrar diseño y contenido.....	5
Adaptación de la aplicación al medio	5
Accesibilidad de la aplicación	5
2.2 Modelo-Vista-Controlador	6
¿Qué es MVC?.....	6
Ciclo de vida del MVC.....	6
Ventajas de MVC.....	7
Desventajas de MVC	7
Entornos de trabajo PHP bajo el patrón arquitectónico MVC	7
Zend Framework.....	7
CakePHP.....	8
CodeIgniter	8
2.3 Lenguajes	8

HTML5	8
Estructura global	9
Organización y nuevos elementos HTML5	10
Hojas de estilo en cascada	16
Selectores	16
Pseudo-elementos	17
Bordes	17
Fondos	18
Efectos de texto	18
Transiciones	18
JQuery	19
SQL	20
Características y ventajas de SQL	21
CAPÍTULO 3 METODOLOGÍA	22
Metodología para el desarrollo de software	23
CAPÍTULO 4 DESARROLLO.....	25
4.1 Análisis de requerimientos	26
Diagrama de caso de uso del negocio actual	26
Diagramas de caso de uso de la aplicación	27
Diagrama de caso de uso del asistente.....	27
Diagrama de caso de uso del profesor	30
Diagrama de caso de uso del alumno	30
Diagrama de caso de uso del administrador	31
4.2 Diseño	33
Mapa de navegación	33
Estructura o <i>layout</i>	35
Diseño de la base de datos	36
4.3 Codificación.....	37
Configuración de la aplicación	39
Modelo – Vista – Controlador	41
Modelo encargado del módulo de inicio de sesión	41
Vista del módulo de inicio de sesión	43
Controlador del módulo de inicio de sesión	44
Directorios adicionales	45
4.4 Pruebas	46
Solicitud de registro de proyecto de titulación.....	46
Registro en el sistema.....	46
Impresión de oficio de registro.....	50

Impresión de oficio del Área Académica.....	52
Aprobación del trabajo e impresión del oficio	55
Aprobación de trabajos	55
Impresión de oficio de aprobación	58
Usabilidad del sistema	61
CONCLUSIÓN.....	62
TRABAJO FUTURO.....	64
BIBLIOGRAFÍA	66
ANEXO A.....	68
Requerimientos funcionales	69
ANEXO B.....	80
Código fuente de la aplicación	81
ANEXO C.....	82
Manuales de usuario.....	83
ANEXO D.....	84
Cuestionario usabilidad (asistentes)	85

Índice de figuras.

Figura 1. Ciclo de vida del MVC.	7
Figura 2. Estructura clásica de un sitio web.	11
Figura 3. Estructura básica de un sitio web HTML5.	12
Figura 4. Uso de SQL para acceder a la base de datos.	20
Figura 5. Ciclo de vida del software.....	23
Figura 6. Caso de uso del negocio actual.....	26
Figura 7. Caso de uso general de la aplicación.	27
Figura 8. Caso de uso del actor Asistente (gestión de usuarios).	28
Figura 9. Caso de uso del actor Asistente (gestión de trabajos).	29
Figura 10. Caso de uso del actor Profesor.	30
Figura 11. Caso de uso del actor Alumno.	31
Figura 12. Caso de uso del actor Administrador.	32
Figura 13. Mapa de navegación del administrador.	33
Figura 14. Mapa de navegación del/la asistente.	33
Figura 15. Mapa de navegación del profesor.	34
Figura 16. Mapa de navegación del alumno sin trabajo registrado.	34
Figura 17. Mapa de navegación del alumno con trabajo pendiente.	34
Figura 18. Mapa de navegación del alumno con trabajo aprobado.....	34
Figura 19. Resolución de pantalla.....	35
Figura 20. Diseño de la estructura de la aplicación.	36
Figura 21. Diagrama E-R de la base de datos.	37
Figura 22. Diagrama de flujo de Codeigniter.....	37
Figura 23. Directorio raíz de la aplicación.	38
Figura 24. Directorio application.	39
Figura 25. Directorio config.	39
Figura 26. Solicitud de registro de proyecto de titulación.	48
Figura 27. Registro de una Monografía.	49
Figura 28. Oficio de registro de una Tesis.	50
Figura 29. Oficio de registro de una Monografía.....	51
Figura 30. Generar Oficio del Área Académica.....	52
Figura 31. Oficio de la academia (tesis).	53
Figura 32. Oficio de la academia (monografía).....	54
Figura 33. Trabajo registrado (ventana principal del asistente).....	55
Figura 34. Detalles del registro de un trabajo de tesis.	56
Figura 35. Detalle del registro de un trabajo monográfico.	57
Figura 36. Oficio de respuesta a trabajo de tesis.	59
Figura 37. Oficio de respuesta a trabajo monográfico.	60

Índice de tablas

Tabla 1. Doctype HTML5.....	9
Tabla 2. Estructura de un documento HTML5 etiqueta <html>.....	9
Tabla 3. Estructura de un documento HTML5 etiqueta <head>.....	9
Tabla 4. Estructura de un documento HTML5 etiqueta <body>.....	10
Tabla 5. Estructura de un documento HTML5 etiqueta <header>.....	12
Tabla 6. Estructura de un documento HTML5 etiqueta <nav>.....	13
Tabla 7. Estructura de un documento HTML5 etiqueta <section>.....	14
Tabla 8. Estructura de un documento HTML5 etiqueta <aside>.....	15
Tabla 9. Estructura de un documento HTML5 etiqueta <footer>.....	16
Tabla 10. Selectores de atributos CSS3.....	17
Tabla 11. Selectores de hermanos CSS3.....	17
Tabla 12. Pseudo-elementos CSS3.....	17
Tabla 13. Bordes redondeados CSS3.....	17
Tabla 14. Sombras CSS3.....	17
Tabla 15. Bordes con imágenes CSS3.....	17
Tabla 16. Tamaño de la imagen CSS3.....	18
Tabla 17. Origen de la imagen CSS3.....	18
Tabla 18. Múltiples imágenes de fondo CSS3.....	18
Tabla 19. Efectos de texto CSS3.....	18
Tabla 20. Transiciones CSS3.....	19
Tabla 21. JQuery utilizando un CDN de JQuery.....	19
Tabla 22. JQuery utilizando un el CDN de Google.....	19
Tabla 23. JQuery vinculado desde el servidor.....	20
Tabla 24. Configuración de config.php.....	40
Tabla 25. Configuración de routes.php (default controller).....	40
Tabla 26. Configuración de routes.php (uri routing).....	40
Tabla 27. Configuración de database.php.....	41
Tabla 28. Modelo del asistente.....	42
Tabla 29. Cargar el modelo.....	43
Tabla 30. Referencia a una función del modelo.....	43
Tabla 31. Vista del módulo de inicio de sesión.....	44
Tabla 32. Cargar la vista del módulo de sesión.....	44
Tabla 33. Controlador login.....	45
Tabla 34. Cargar una librería o helper.....	46
Tabla 35. Uso de las funciones de una librería.....	46
Tabla 36. Uso de las funciones de un helper.....	46
Tabla 37. Especificación del requerimiento autenticarse en el sistema (alumno).....	69
Tabla 38. Especificación del requerimiento registro del trabajo.....	69
Tabla 39. Especificación del requerimiento generar oficios (alumno).....	70
Tabla 40. Especificación del requerimiento editar trabajo (alumno).....	70
Tabla 41. Especificación del requerimiento autenticarse en el sistema (asistente).....	71
Tabla 42. Especificación del requerimiento crear usuarios.....	71
Tabla 43. Especificación del requerimiento editar usuarios.....	72
Tabla 44. Especificación del requerimiento asignar división (asistente).....	72

Tabla 45. Especificación del requerimiento valida trabajo.....	72
Tabla 46. Especificación del requerimiento eliminar trabajos.	73
Tabla 47. Especificación del requerimiento habilitar edicion de trabajos.....	73
Tabla 48. Especificación del requerimiento editar trabajos.	73
Tabla 49. Especificación del requerimiento envío de notificaciones.....	74
Tabla 50. Especificación del requerimiento generar oficios (asistente).	74
Tabla 51. Especificación del requerimiento generar reportes (asistente).....	74
Tabla 52. Especificación del requerimiento asignar Programa Educativo.....	75
Tabla 53. Especificación del requerimiento consultar registros (asistente).....	75
Tabla 54. Especificación del requerimiento agregar egresados.	75
Tabla 55. Especificación del requerimiento autenticarse en el sistema (profesor).	76
Tabla 56. Especificación del requerimiento generar reportes (profesor).	76
Tabla 57. Especificación del requerimiento generar oficios (profesor).....	76
Tabla 58. Especificación del requerimiento consultar registros (profesor).....	77
Tabla 59. Especificación del requerimiento autenticarse en el sistema (administrador).	77
Tabla 60. Especificación del requerimiento crear usuarios (administrador).....	77
Tabla 61. Especificación del requerimiento editar usuarios (administrador).....	78
Tabla 62. Especificación del requerimiento asignar división (administrador).....	78
Tabla 63. Especificación del requerimiento nueva división.....	78
Tabla 64. Especificación del requerimiento nueva carrera.	79
Tabla 65. Especificación del requerimiento agregar director.....	79
Tabla 66. Especificación del requerimiento eliminar director.....	79

CAPÍTULO 1

INTRODUCCIÓN

1.1 Descripción del problema

El Reglamento de Estudios Técnicos y de Licenciatura de la Universidad de Quintana Roo en su título cuarto “De los planes y programas de estudio Capítulo V Titulación profesional” nos muestra las distintas maneras y procesos que se deben llevar a cabo para que un estudiante universitario pueda obtener su título profesional.

La División de Ciencias e Ingeniería (DCI), administra el proceso de titulación de cada alumno de las carreras pertenecientes a esta división, sin embargo no cuenta con un sistema de información que facilite y lleve el control durante el proceso.

La gestión del proceso de titulación es a través del software de ofimática Microsoft Excel Office, donde se encuentran guardados los temas de tesis de los aspirantes, además los documentos involucrados en este proceso se generan manualmente por lo que pueden presentarse las siguientes dificultades:

1. El registro de tesis se encuentra guardado en la PC de la asistente de la Dirección, por lo que esta información es susceptible a todo aquel que pueda tener acceso a la PC.
2. No existe un registro de los trabajos monográficos, debido a que sólo se notifica a la Dirección al momento en que se titulará el egresado.
3. La actualización o modificación del registro de los trabajos monográficos y de tesis, se realizan manualmente en un archivo de Microsoft Excel Office por parte de la asistente de Dirección, por lo que se pueden presentar confusiones en el registro y no existe un historial en caso de cualquier cambio en el comité de titulación o monografía.
4. La falta de una base de datos, dificulta la búsqueda de algún trabajo en específico, así como la posibilidad de emitir reportes por programa educativo, año de titulación o tipo de titulación.
5. Cada documento emitido tanto para el alumno como para los integrantes del comité de titulación o revisión se realizan manualmente, por lo que si se presenta alguna modificación, se tiene que emitir un nuevo documento, generando pérdida de tiempo en búsqueda de la información y la realización de los oficios.
6. Los reportes emitidos por la asistente de Dirección se realizan de forma manual, por lo que puede generar un atraso en su entrega.
7. Es tedioso hacer una relación de trabajos de titulación por programa educativo o por integrante de un comité de titulación, cuando esta estadística es requerida.

1.2 Objetivos

Objetivo general

Desarrollar un sistema basado en web para la administración de tesis y monografías que facilite el registro, actualización y generación de reportes relativos a los trabajos de titulación de los programas educativos en la División de Ciencias e Ingeniería de nivel licenciatura.

Objetivos específicos

- 1) Realizar un análisis de requisitos del proceso de titulación en la DCI.
- 2) Diseñar las interfaces de usuario, la arquitectura del sistema, la base de datos y las pruebas del sistema.
- 3) Configurar el entorno de desarrollo del sistema.
- 4) Configurar un patrón de desarrollo para el sistema.
- 5) Codificar y probar el sistema desarrollado.
- 6) Crear la documentación del usuario.
- 7) Implementación del sistema.

1.3 Justificación

El diseño de un sistema para la administración de los trabajos monográficos y de tesis, proporcionará los siguientes beneficios:

1. Mejorar la seguridad de los datos mediante:
 - a) La administración de usuarios y permisos.
 - b) Creación de una base de datos que almacene la información sobre los trabajos de titulación (tesis o monografías).
 - c) Resguardo de la información en un servidor, al cual sólo tendrá acceso el administrador del sistema.
2. Facilitar la actualización y registro de los trabajos de titulación.
3. Generar los documentos oficiales que se entregan a los aspirantes a titularse y también a los integrantes del comité.
4. Facilitar la generación de reportes estadísticos (profesores y asistentes).
5. Generar un historial de los trabajos de titulación.

1.4 Alcance y restricciones.

Se llevará a cabo la implementación del sistema desarrollado en la División de Ciencias e Ingeniería para las carreras de nivel licenciatura que se encuentran actualmente registradas. El diseño deberá cumplir las características de usabilidad web y utilizar tecnologías como HTML5 y hojas de estilos en cascada (CSS3) entre otras, para asegurar estándares actuales en los navegadores. Para las pruebas del sistema se utilizará información correspondiente al 2014.

CAPÍTULO 2

MARCO TEÓRICO

2.1 Usabilidad de una aplicación web

El término usabilidad es relativamente reciente. Se puede decir que surgió poco después de la expansión de Internet, a finales de los años 90. Pero realmente la idea que quiere expresar es bastante anterior, aunque se conocía con otros nombres como por ejemplo el Diseño Centrado en el Usuario o *Human-Computer Interaction* (HCI).

La usabilidad se refiere a la capacidad de un software o sistema interactivo de ser comprendido, aprendido, usado fácilmente y atractivo para un usuario, en condiciones específicas de uso. También es la efectividad, eficiencia y satisfacción con la que un producto permite alcanzar sus objetivos específicos (1).

Conceptos básicos de usabilidad

Facilidad de manejo de la aplicación

Generalmente el diseñador de una aplicación es un usuario subjetivo del programa, por lo que sabe exactamente cómo manejarlo. El problema de esto, es que en ocasiones no se tiene en cuenta el nivel de experiencia de los usuarios medios.

Para solucionar este problema surge la usabilidad.; como se definió anteriormente, la usabilidad se centra en la facilidad e intuitividad en el manejo de las aplicaciones.

Equilibrar diseño y contenido

En muchas ocasiones los diseñadores inexpertos hacen uso de la regla de oro “una imagen dice más que mil palabras”. Esto no es lo adecuado, ya que en un principio podría lucir fascinante el diseño pero no están considerando la carga de todo el contenido multimedia y las limitaciones de algunos usuarios con la velocidad de internet que poseen.

Adaptación de la aplicación al medio

Uno de los problemas que se tiene que enfrentar cuando se diseña una aplicación, es el entorno en donde se ejecutará ya que no se sabe a ciencia cierta el medio donde se desempeñará.

Para afrontar esto, lo principal es preguntar al cliente el tipo de entorno donde se desarrollará; sin embargo esto no aplica para todas las aplicaciones, ya que en la actualidad y con los avances tecnológicos en hardware y con el creciente uso de dispositivos móviles, es indispensable crear aplicaciones independientes a este.

Accesibilidad de la aplicación

Un último aspecto que hace a una aplicación usable es su accesibilidad para personas con necesidades diferentes. Hemos de centrar especialmente nuestra atención en estos usuarios y nuestra aplicación deberá adoptar ciertas medidas para que puedan utilizarla. Algunas de las medidas que se pueden tomar para alcanzar este objetivo son:

- Utilizar una correcta combinación de colores.
- Dar la opción de aumentar el tamaño de la letra o algún tipo de zoom en la aplicación.

- Proporcionar la opción de imprimir o guardar en disco toda la información que ofrece la aplicación.

2.2 Modelo-Vista-Controlador

¿Qué es MVC?

MVC (*Model-View-Controller* por sus siglas en Inglés) es un patrón de diseño de software en torno a la interconexión de los tres tipos de componentes principales en un lenguaje de programación como PHP, a menudo con un fuerte enfoque en la programación orientada a objetos (POO). Estos tres tipos de componentes son vagamente llamados modelos, vistas y controladores.

El modelo es donde se guarda toda la lógica del negocio de una aplicación, la lógica del negocio puede ser cualquier cosa específica acerca de cómo una aplicación almacena los datos, o utiliza servicios de terceros con el fin de cumplir con sus necesidades. Si la aplicación debe acceder a la información en una base de datos, el código deberá estar guardado en el modelo.

La vista es donde se encontrarán todos los elementos de la interfaz de usuario de una aplicación, esta puede contener código HTML, hojas de estilo CSS y archivos JavaScript. Cualquier cosa que el usuario pueda ver, es guardado en la vista, y algunas veces lo que ve el usuario actualmente es la combinación de varias vistas en la misma petición.

El controlador es el componente encargado de conectar el modelo con la vista. Los controladores aíslan la lógica del negocio de un modelo de los elementos de la interfaz de usuario de una vista y maneja la forma en la que la aplicación responde a la interacción del usuario en la vista. Los controladores son el primer punto de entrada en estos componentes, ya que la primera solicitud se pasa a un controlador, que luego instancia a los modelos y vistas requeridas para cumplir con una petición a la aplicación (2).

Ciclo de vida del MVC.

1. El usuario realiza una petición.
2. El controlador captura la petición del usuario.
3. El controlador llama al modelo.
4. El modelo interactúa con la base de datos, y retorna la información al controlador.
5. El controlador recibe la información y la envía a la vista.
6. La vista procesa la información recibida y la entrega de una manera visualmente entendible al usuario (3).

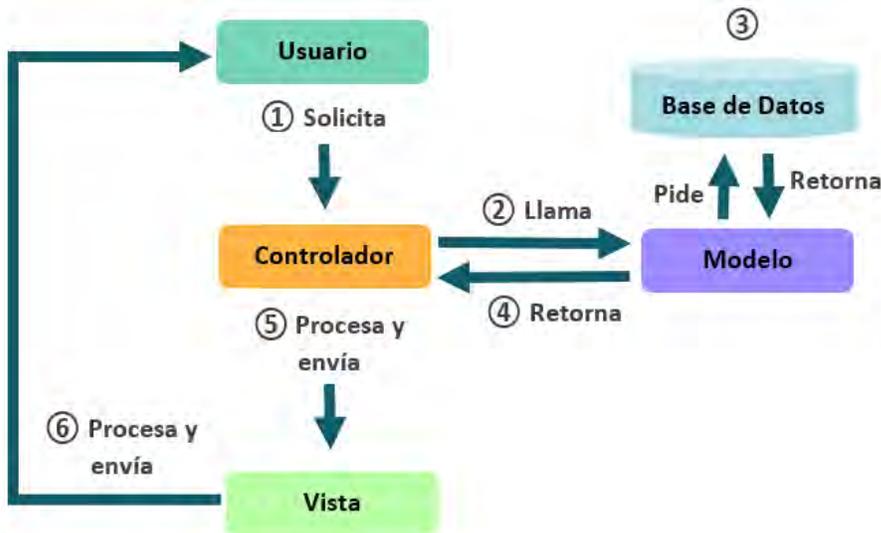


Figura 1. Ciclo de vida del MVC.

Ventajas de MVC

Las principales ventajas del uso del patrón MVC son (4):

1. La separación del *Modelo* y la *Vista*, lo cual logra separar los datos, de su representación visual.
2. Facilita el manejo de errores.
3. Permite que el sistema sea escalable si es requerido.
4. Es posible agregar múltiples representaciones de los datos.

Desventajas de MVC

Las principales desventajas del uso del patrón MVC son (4):

1. La cantidad de archivos que se deben mantener se incrementa considerablemente.
2. La curva de aprendizaje es más alta que utilizando otros modelos.
3. Su separación en capas, aumenta la complejidad del sistema.

Entornos de trabajo PHP bajo el patrón arquitectónico MVC

Zend Framework

Zend *Framework2* (ZF2) es un entorno de trabajo de código abierto para el desarrollo de aplicaciones y servicios web usando PHP 5.3+.

La estructura de componentes de Zend *Framework 2* es único; cada componente está diseñado con pocas dependencias de otros componentes. ZF2 sigue el principio orientado a objetos de diseño sólido. Esta arquitectura de acoplamiento flexible permite a los desarrolladores utilizar cualquier componente que quieran. A esto se le llama "usar a voluntad" el diseño.

Mientras que pueden ser utilizados por separado, los componentes en la biblioteca estándar de Zend *Framework 2* forman un entorno de trabajo de aplicación web potente y flexible cuando se combinan. Además, ofrece una aplicación robusta y de alto rendimiento MVC, una abstracción de base de datos que es fácil de usar, y elementos de formularios HTML5 que renderizan, validan y filtran, de modo que los desarrolladores pueden consolidar todas estas operaciones utilizando una interfaz orientada a objetos fácil de utilizar.

El patrocinador principal del proyecto 'Zend *Framework 2*' es Zend Technologies, pero muchas compañías han contribuido con componentes o características importantes para el entorno de trabajo. Empresas como Google, Microsoft y Strikelron se han asociado con Zend para proporcionar interfaces de servicios web y otras tecnologías que desean poner a disposición de los desarrolladores de Zend *Framework 2* (5).

CakePHP

El entorno de trabajo CakePHP proporciona una base sólida para una aplicación. Se puede manejar todos los aspectos, desde la solicitud inicial del usuario hasta la representación final de una página web. Y puesto que el entorno de trabajo sigue los principios de MVC (*Model-View-Controller*), que permite fácilmente personalizar y extender la mayoría de los aspectos de una aplicación.

El entorno de trabajo también proporciona una estructura organizativa básica, desde los nombres de archivo hasta los nombres de las tablas de la base de datos, manteniendo toda la aplicación consistente y lógica. Este concepto es simple pero poderoso; sigue las convenciones y siempre se sabe exactamente dónde están las cosas y cómo están organizadas (6).

CodeIgniter

CodeIgniter es un entorno de trabajo de desarrollo de aplicaciones - un conjunto de herramientas - para las personas que construyen sitios web utilizando PHP. Su objetivo es permitir el desarrollo de proyectos mucho más rápido de lo que podría si estuviera escribiendo código desde cero, proporcionando un rico conjunto de librerías para tareas comúnmente necesarias, así como una interfaz sencilla y estructura lógica para acceder a estas bibliotecas. *CodeIgniter* permite centrarse en un proyecto, minimizando la cantidad de código necesario para una tarea determinada.

CodeIgniter fue originalmente desarrollado por Rick Ellis (CEO de EllisLab, Inc.). El *framework* fue escrito para el desempeño en el mundo real, con muchas de las bibliotecas de clases (*libraries*), ayudantes (*helpers*), y subsistemas tomados de la base de código de *ExpressionEngine*¹ (7).

2.3 Lenguajes

HTML5

HTML5 no es una nueva versión del antiguo lenguaje de etiquetas, ni siquiera una mejora de esta ya antigua tecnología, sino un nuevo concepto para la construcción de sitios web y aplicaciones en una era que combina dispositivos móviles, computación en la nube y trabajos en red.

¹ <https://ellislab.com/expressionengine>.

HTML5 provee básicamente tres características: estructura, estilo y funcionalidad. No fue declarado oficialmente pero, incluso cuando algunas APIs (Interface de Programación de Aplicaciones) y la especificación de CSS3 por completo no son parte del mismo, HTML5 es considerado el producto de la combinación de HTML, CSS y JavaScript. Estas tecnologías son altamente dependientes y actúan como una sola unidad organizada bajo la especificación de HTML5 (8).

Estructura global

<!DOCTYPE>

Indica el tipo de documento que se va a crear, en HTML5 es extremadamente sencillo, basta con escribir la siguiente etiqueta:

```
<!DOCTYPE html>
```

Tabla 1. Doctype HTML5.

Esta línea debe ser la primera línea del archivo, sin espacios o líneas que la precedan. De esta forma, el modo estándar del navegador es activado y las incorporaciones de HTML5 son interpretadas siempre que sea posible, o ignoradas en caso contrario.

<html>

Luego de declarar el tipo de documento, se debe comenzar a construir la estructura HTML.

```
<!DOCTYPE html>
<html lang="es">
</html>
```

Tabla 2. Estructura de un documento HTML5 etiqueta <html>.

El atributo *lang* en la etiqueta de apertura <html> es el único atributo que se necesita especificar en HTML5, este atributo define el idioma humano del contenido del documento que se está creando.

<head>

El código HTML insertado en las etiquetas <html> está dividido en dos secciones principales. La primera de ellas es la cabecera <head> y la segunda el cuerpo <body>.

```
<!DOCTYPE html>
<html lang="es">
<head>
</head>
</html>
```

Tabla 3. Estructura de un documento HTML5 etiqueta <head>.

Dentro de las etiquetas <head> definimos el título de nuestra página web, se declara el set de caracteres correspondiente, se provee información general acerca del documento y se incorporan los archivos externos de hojas de estilos (CSS) y códigos de JavaScript.

Excepto por el título y algunos iconos, el resto de la información incorporada en el documento entre estas etiquetas es invisible para el usuario.

`<body>`

El cuerpo representa la parte visual de todo el documento y es especificado entre las etiquetas `<body>`.

```
<!DOCTYPE html>
<html lang="es">
<head>
</head>
<body>
</body>
</html>
```

Tabla 4. Estructura de un documento HTML5 etiqueta `<body>`.

Organización y nuevos elementos HTML5

La Figura 2 representa la estructura clásica de un sitio web, en donde se observan las siguientes secciones:

1. **Cabecera:** espacio donde usualmente se coloca el logo y el título del sitio web.
2. **Barra de Navegación:** espacio destinado para el menú de navegación del sitio web.
3. **Información Principal:** en esta sección se coloca todo el contenido relevante del sitio.
4. **Barra Lateral:** esta sección está encargada de mostrar información complementaria, como pueden ser enlaces externos, o información del autor de alguna nota.
5. **Pie de Página:** aquí se coloca información de la institución o empresa, términos y condiciones y los derechos de autor.



Figura 2. Estructura clásica de un sitio web.

HTML5 considera esta estructura y añade nuevos elementos los cuales representan a cada una de estas secciones.

La Figura 3 muestra el diseño clásico de un sitio web, pero esta vez representado por los nuevos elementos HTML5.

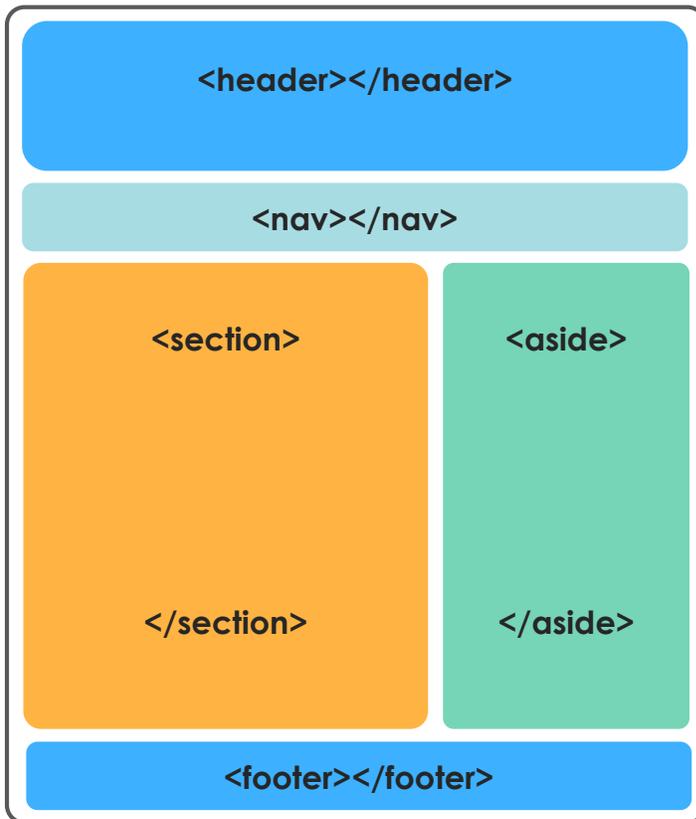


Figura 3. Estructura básica de un sitio web HTML5.

`<header>`

Provee información introductoria como pueden ser logos, títulos, subtítulos. Sin embargo no se debe confundir con respecto a la etiqueta `<head>` ya que esta etiqueta está encargada de mostrar información acerca de todo el documento, mientras de `<header>` es utilizada únicamente para el cuerpo del documento.

```

<!DOCTYPE html>
<html lang="es">
<head>
  <tittle>Titulo del documento</tittle>
</head>
<body>
  <header>
    <h1>Titulo principal del sitio web</h1>
  </header>
</body>
</html>

```

Tabla 5. Estructura de un documento HTML5 etiqueta `<header>`.

<nav>

Esta nueva etiqueta HTML5 representa a la **Barra de Navegación**.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <tittle>Titulo del documento</tittle>
</head>
<body>
  <header>
    <h1>Titulo principal del sitio web</h1>
  </header>
  <nav>
    <ul>
      <li>Opción 1</li>
      <li>Opción 2</li>
      <li>Opción 3</li>
    </ul>
  </nav>
</body>
</html>
```

Tabla 6. Estructura de un documento HTML5 etiqueta <nav>.

<section>

La etiqueta <section> es la encargada de representar a la **Información Principal**, en ella estará toda la información relevante del sitio web, debido a que la información puede estar dividida en bloques o columnas, el propósito de esta etiqueta HTML5 es especificar dichas secciones de una manera más sencilla.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <tittle>Título del documento</tittle>
</head>
<body>
  <header>
    <h1>Título principal del sitio web</h1>
  </header>
  <nav>
    <ul>
      <li>Opción 1</li>
      <li>Opción 2</li>
      <li>Opción 3</li>
    </ul>
  </nav>
  <section>

  </section>
</body>
</html>
```

Tabla 7. Estructura de un documento HTML5 etiqueta <section>.

<aside>

En la estructura clásica de un sitio web (Figura 2) la parte correspondiente a la **Barra Lateral** se encarga de mostrar datos relacionados a la información principal pero con menor relevancia.

La etiqueta <aside> será la encargada de representar dicha sección.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <tittle>Título del documento</tittle>
</head>
<body>
  <header>
    <h1>Título principal del sitio web</h1>
  </header>
  <nav>
    <ul>
      <li>Opción 1</li>
      <li>Opción 2</li>
```

```

        <li>Opción 3</li>
      </ul>
    </nav>
  <section>

  </section>
  <aside>
    <p>Mensaje 1</p>
    <p>Mensaje 2</p>
  </aside>
</body>
</html>

```

Tabla 8. Estructura de un documento HTML5 etiqueta <aside>.

<footer>

Por último, la etiqueta <footer> es la encargada de representar al **Pie de Página** de la estructura clásica de un sitio web (Figura 2), en esta etiqueta estará contenida toda la información adicional que el desarrollador desee mostrar.

Este elemento al igual que el <header>, puede ser repetido a lo largo del documento, representando el final de cada sección.

```

<!DOCTYPE html>
<html lang="es">
<head>
  <title>Título del documento</title>
</head>
<body>
  <header>
    <h1>Título principal del sitio web</h1>
  </header>
  <nav>
    <ul>
      <li>Opción 1</li>
      <li>Opción 2</li>
      <li>Opción 3</li>
    </ul>
  </nav>
  <section>

  </section>
  <aside>

```

```

    <p>Mensaje 1</p>
    <p>Mensaje 2</p>
  </aside>
  <footer>
    <p>Derechos reservados &copy; 2014-2015</p>
  </footer>
</body>
</html>

```

Tabla 9. Estructura de un documento HTML5 etiqueta <footer>.

Hojas de estilo en cascada

Las hojas de Estilo en Cascada (CSS) se refiere a un lenguaje sencillo que define los estilos que se pueden aplicar a HTML. Cuando HTML describe la estructura de una página web, CSS describe su presentación.

El objetivo principal de CSS es separar la estructura (HTML) de la presentación (CSS).

Separar la estructura de la presentación tiene numerosas ventajas:

- Una hoja de estilos en cascada, puede ser compartida a través de múltiples páginas web.
- Los sitios web son más fáciles de mantener y se vuelven más flexibles.
- Los estilos aplicados a una página web pueden ser adaptados a varios dispositivos y entornos.

En la web moderna, el papel de CSS es más importante que nunca. Como la tecnología ha avanzado y las formas en que se accede a la web son más diversas, CSS debe evolucionar para adaptarse tanto a los usuarios que desean acceder a la información a través de la web y los desarrolladores que desean presentar la información de una manera particular (9).

Algunos de los módulos CSS3 más importantes son:

Selectores

Son herramientas que sirven para seleccionar uno o varios elementos del código HTML.

Selectores de atributos:

```

/* Selecciona todos los enlaces que lleven a una página que contenga la
palabra ejemplo.*/
a[href*="ejemplo"] {...}
/*Selecciona todos los elementos que apunten a una dirección de correo*/
a[href^="mailto:"] {...}
/* Selecciona todos los elementos que apuntan a páginas PHP.*/
a[href$=".php"] {...}

```

Tabla 10. Selectores de atributos CSS3.

Selectores de hermanos:

Selecciona el elemento *h2* que se encuentra después de un elemento *h1*

```
h1 ~ h2{...}
```

Tabla 11. Selectores de hermanos CSS3.

Pseudo-elementos

Los pseudo-elementos en CSS3 permanecen con respecto a CSS2.1 sin embargo la sintaxis cambia de “:” a “::”.

Selection es un nuevo pseudo-elemento que incorpora CSS3 el cual permite que al seleccionar el texto se pueda cambiar el color de fondo y el color del texto.

```
P::selection{...}
```

Tabla 12. Pseudo-elementos CSS3.

Bordes

Se pueden crear bordes redondeados, agregar sombras a los bordes y utilizar imágenes de bordes.

Bordes redondeados

```
div {
  border: 2px solid;
  border-radius: 25px;
}
```

Tabla 13. Bordes redondeados CSS3.

Sombras

```
div {
  box-shadow: 10px 10px #888888;
}
```

Tabla 14. Sombras CSS3.

Bordes con imágenes

```
div {
  -webkit-border-image: url(border.png) 30 30 round;
  -o-border-image: url(border.png) 30 30 round;
  border-image: url(border.png) 30 30 round;
}
```

Tabla 15. Bordes con imágenes CSS3.

Fondos

CSS3 incorpora nuevas propiedades para los fondos, como pueden ser el tamaño de la imagen, el origen o posición y múltiples imágenes de fondo.

Tamaño de la imagen

```
div {  
    background: url(img_flwr.gif);  
    background-size: 80px 60px;  
    background-repeat: no-repeat;  
}
```

Tabla 16. Tamaño de la imagen CSS3.

Origen de la imagen

```
div {  
    background: url(img_tree.gif), url(img_flwr.gif);  
    background-size: 100% 100%;  
    background-repeat: no-repeat;  
    background-origin: content-box;  
}
```

Tabla 17. Origen de la imagen CSS3.

Múltiples imágenes de fondo

```
div {  
    background: url(img_tree.gif), url(img_flwr.gif);  
    background-size: 100% 100%;  
    background-repeat: no-repeat;  
}
```

Tabla 18. Múltiples imágenes de fondo CSS3.

Efectos de texto

CSS3 Incorpora las sombras en los textos.

```
h1 {  
    text-shadow: 5px 5px 5px #FF0000;  
}
```

Tabla 19. Efectos de texto CSS3.

Transiciones

Permite que los elementos cambien de un estilo a otro gradualmente.

```
div {  
    -webkit-transition: width 2s;
```

```
transition: width 2s;  
}
```

Tabla 20. Transiciones CSS3.

JQuery

jQuery es una biblioteca JavaScript que intenta hacer la programación de JavaScript más fácil y divertida. Una biblioteca de JavaScript es un complejo conjunto de código JavaScript que tanto simplifica las tareas difíciles como resuelve problemas cross-browser. En otras palabras, jQuery resuelve los dos mayores dolores de cabeza de JavaScript: la complejidad y la compatibilidad con los distintos navegadores.

Con jQuery, se pueden realizar tareas en una sola línea de código que podría tomar cientos de líneas de programación y muchas horas de pruebas en navegadores para lograr con código JavaScript puro.

jQuery es simplemente un conjunto de código JavaScript en un archivo externo JavaScript. Al igual que cualquier archivo JavaScript externo, es necesario vincularlo a la página web. Sin embargo, tiene algunas opciones cuando se trata de añadir a una página web. Se puede utilizar una versión alojada en Google, Microsoft, o jQuery.com, o se puede descargar el archivo de jQuery propio y añadirlo a la página web.

El primer método utiliza una CDN (red de distribución de contenido), es decir otro sitio web que almacena el archivo JQuery y lo envía a cualquiera que lo solicite (10).

Utilizando un el CDN de JQuery.

```
<!DOCTYPE html>  
<html lang="es">  
<head>  
  <script src="http://code.jquery.com/jquery-1.11.2.min.js"></script>  
</head>  
</html>
```

Tabla 21. JQuery utilizando un CDN de JQuery.

Utilizando el CDN de Google.

```
<!DOCTYPE html>  
<html lang="es">  
<head>  
  <script  
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>  
</head>  
</html>
```

Tabla 22. JQuery utilizando un el CDN de Google.

El segundo método para vincular el archivo JQuery, es descargándolo y alojándolo en nuestro servidor y acceder a él mediante la ruta de su ubicación.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <script src="js/jquery.min.js"></script>
</head>
</html>
```

Tabla 23. JQuery vinculado desde el servidor.

SQL

El lenguaje SQL y los sistemas de base de datos relacionales basados en él son una de las tecnologías básicas más importantes de la industria informática.

SQL es una herramienta para la organización, gestión y recuperación de los datos almacenados en bases de datos informáticas.

El programa informático que controla la base de datos se denomina *sistema gestor de base de datos (database management system)*, o SGBD.

Cuando se necesita recuperar datos de una base de datos, se utiliza el lenguaje SQL para formular la solicitud. El SGBD procesa la solicitud de SQL, recupera los datos solicitados y los devuelve al usuario. Este proceso de solicitud de datos de las bases de datos y de recepción de los resultados se denomina *consulta* de la base de datos (11).

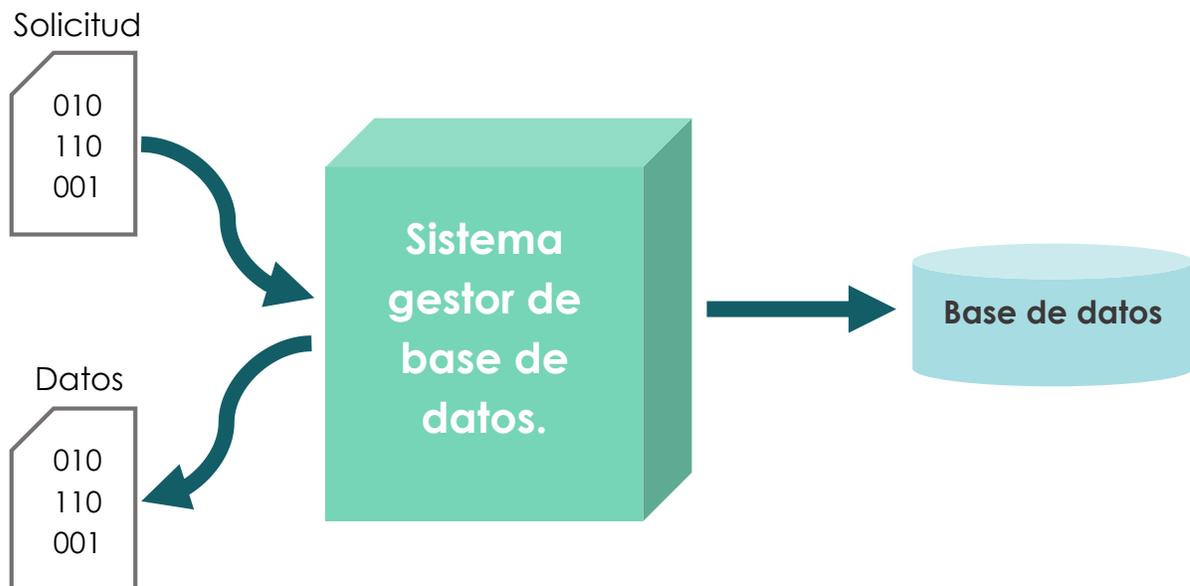


Figura 4. Uso de SQL para acceder a la base de datos.

Características y ventajas de SQL

SQL es un lenguaje sencillo de comprender y una herramienta completa para la administración de datos, la cual posee las siguientes ventajas:

- Independencia del fabricante.
- Transportabilidad.
- Estándares SQL.
- Fundamentos relacionales.
- Estructura de alto nivel en inglés.
- Vistas múltiples de datos.
- Lenguaje completo de base de datos.
- Arquitectura cliente/servidor.

CAPÍTULO 3

METODOLOGÍA

Metodología para el desarrollo de software

El método de investigación y de desarrollo a utilizar, será la metodología para el desarrollo de software.

El desarrollo de un buen sistema de software, se realiza durante el ciclo de vida, que es el periodo de tiempo que se extiende desde la concepción inicial del sistema hasta su eventual retirada de la comercialización o uso del mismo.

El ciclo de vida es un proceso iterativo, de modo que se modificarán las sucesivas etapas en función de las especificaciones de los requisitos producidos en la fase de diseño o implementación, o bien una vez que el sistema se ha implementado, y probado, pueden aparecer errores, que será necesario corregir y depurar y que requieren la repetición de etapas anteriores (12).

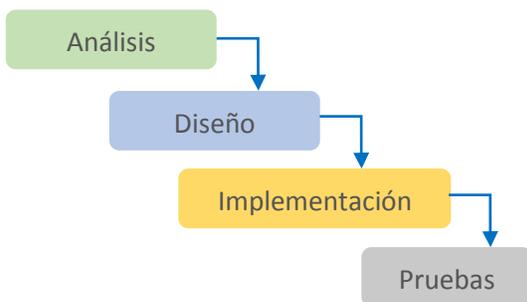


Figura 5. Ciclo de vida del software.

Análisis

La primera etapa del desarrollo de software (análisis), se divide en dos fases:

- Análisis y definición del problema: analizar los requerimientos del usuario.
- Especificaciones de requisitos: en esta etapa, puede ser de utilidad la creación de un prototipo final del sistema, esto para darle al usuario una idea de lo que el sistema será, además de que en esta etapa se pueden hacer anotaciones de los cambios que el usuario quiera realizar y de esta manera no realizar modificaciones en etapas posteriores, lo cual podría complicar el desarrollo de este mismo.

El resultado final de la fase de análisis, es un documento de especificaciones de los requisitos del software, este documento describe explícitamente la funcionalidad del producto, es decir lo que se supone que hace el producto.

Diseño

Esta etapa consta de los siguientes puntos:

- Estructura de la aplicación: de qué manera estará estructurada la navegación del sitio.
- Diseño de interfaces: aspecto visual de la aplicación.

- Diseño de la base de datos: se especifican las entidades y las relaciones que existen entre estas.

La función principal de esta etapa es tener una visión clara de lo que será el sistema visualmente. Otro de sus propósitos es poder realizar modificaciones sin tener que afectar la etapa de codificación.

Implementación

En la etapa de implementación o codificación, se traduce todo lo especificado en etapas anteriores (descripción de lo que es el sistema) en un lenguaje de programación, este lenguaje deberá ser capaz de cumplir con las especificaciones que el sistema requiere.

Pruebas

La etapa de pruebas, como su nombre lo indica, es la prueba y verificación de la funcionalidad del programa. Cualquier error que se encuentre durante esta prueba se debe corregir.

CAPÍTULO 4

DESARROLLO

4.1 Análisis de requerimientos

La fase inicial de la construcción de la aplicación, es la obtención de los requerimientos.

Inicialmente se realizó la visita con la asistente encargada de la gestión de los trabajos de titulación con el fin conocer la manera en la cual se lleva a cabo el registro y la administración de los trabajos de titulación.

Diagrama de caso de uso del negocio actual

Como primera instancia, se obtuvo el diagrama de caso de uso del negocio actual.

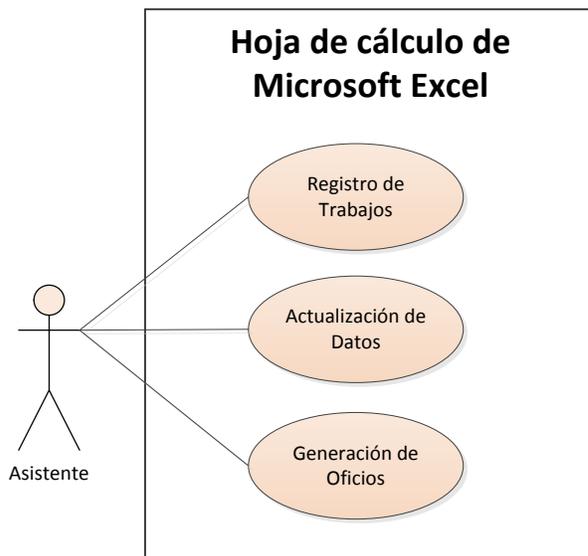


Figura 6. Caso de uso del negocio actual.

En la *Figura 6*, podemos observar la forma en la cual se lleva a cabo la gestión de trabajos de titulación en la actualidad; este método podría parecer a simple vista una manera sencilla de administrar los trabajos, sin embargo, no es el adecuado ya que carece de profesionalidad y es tedioso.

Posteriormente, nos dimos a la tarea de involucrar a más de un usuario en el nuevo sistema y con esto asignarle a cada uno de ellos las tareas que vayan de acuerdo con su participación en todo el proceso.

En la *Figura 7* podemos observar de forma general la participación de cada actor (usuario) en el sistema.

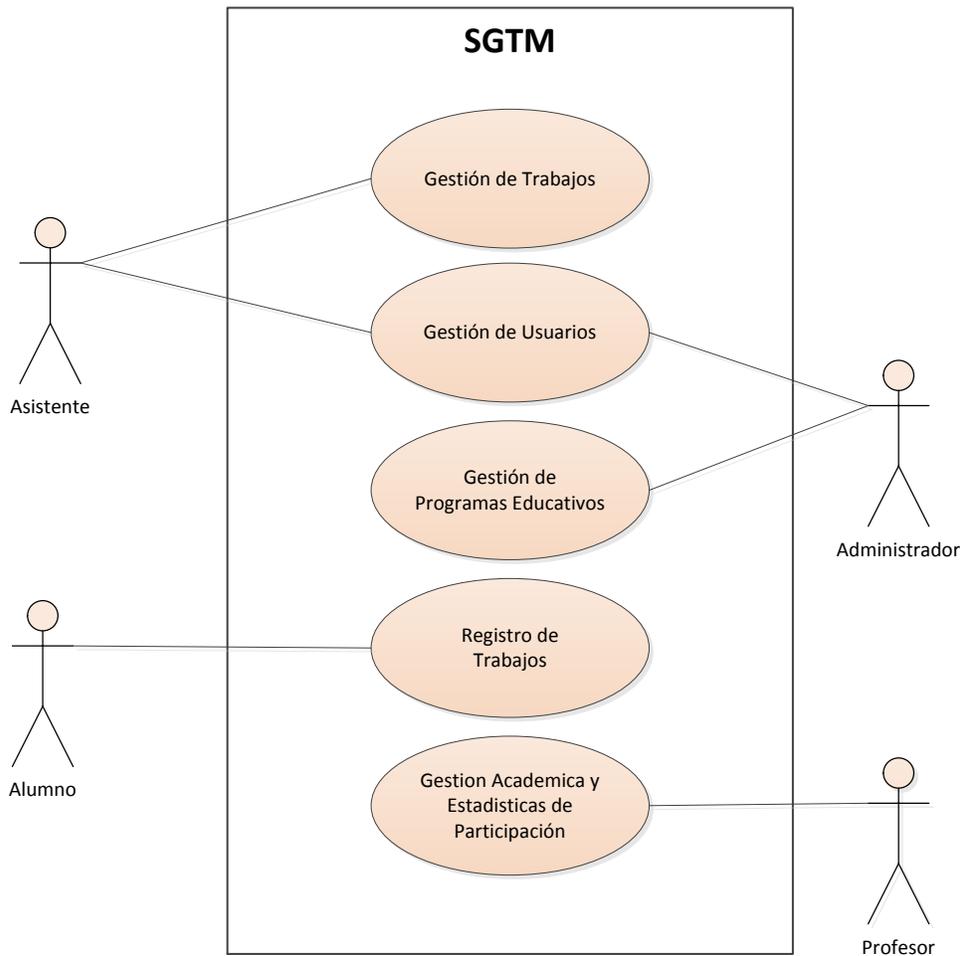


Figura 7. Caso de uso general de la aplicación.

Para tener una idea más clara de las funciones que pueden realizar los usuarios de la aplicación, se analizaron cada uno de los requerimientos funcionales por cada usuario, véase [Anexo A](#)

Posteriormente se procedió a realizar los diagramas de caso de uso de la aplicación, los cuales describen gráficamente las funciones que podrán realizar los usuarios.

Diagramas de caso de uso de la aplicación

Ya teniendo una idea general del comportamiento de los usuarios en la aplicación, realizamos los diagramas de caso de uso para cada uno de ellos y de esta manera tener claridad en las acciones que pueden o no realizar.

Diagrama de caso de uso del asistente

En la *Figura 8* y *Figura 9* se muestran sólo los casos de uso del actor asistente y las acciones que son válidas para este tipo de usuario. Para el SGTM (Sistema de Gestión de Tesis y Monografías) el usuario asistente, es el encargado de la gestión de usuarios y trabajos de titulación de la división a la que pertenece.

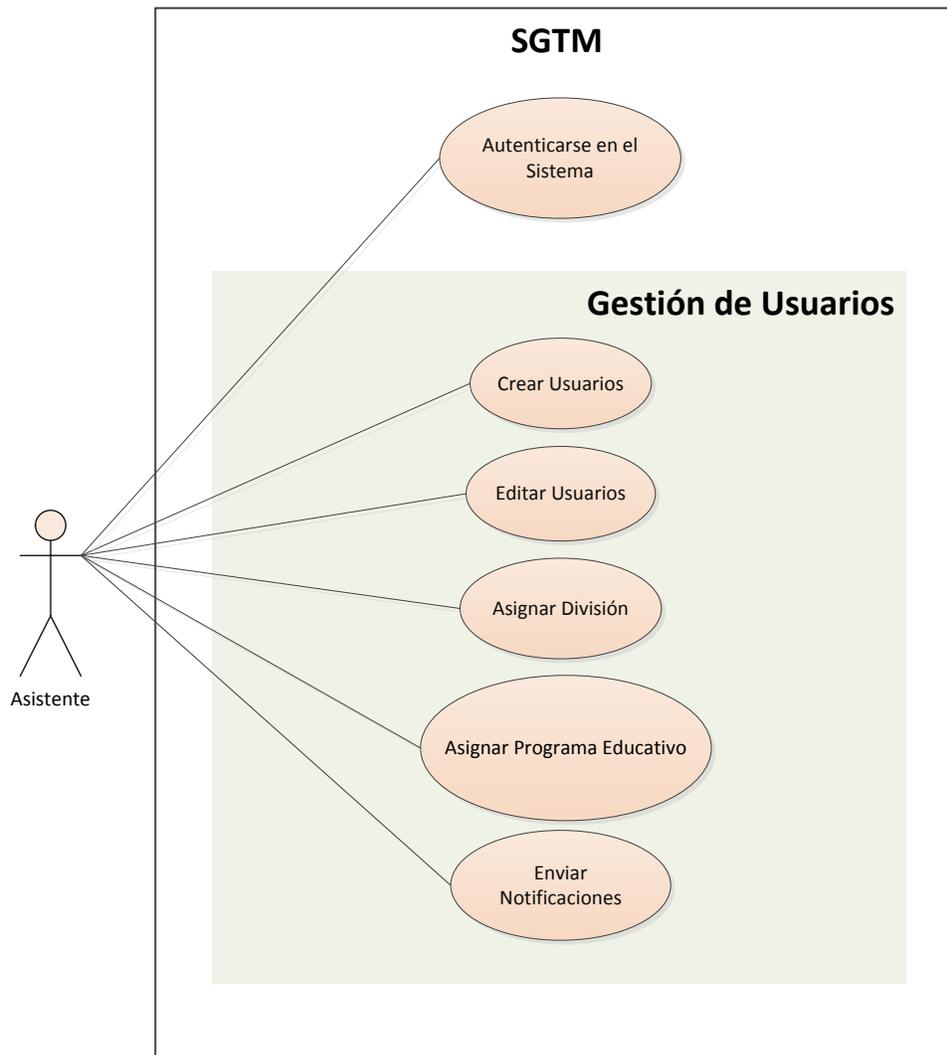


Figura 8. Caso de uso del actor Asistente (gestión de usuarios).

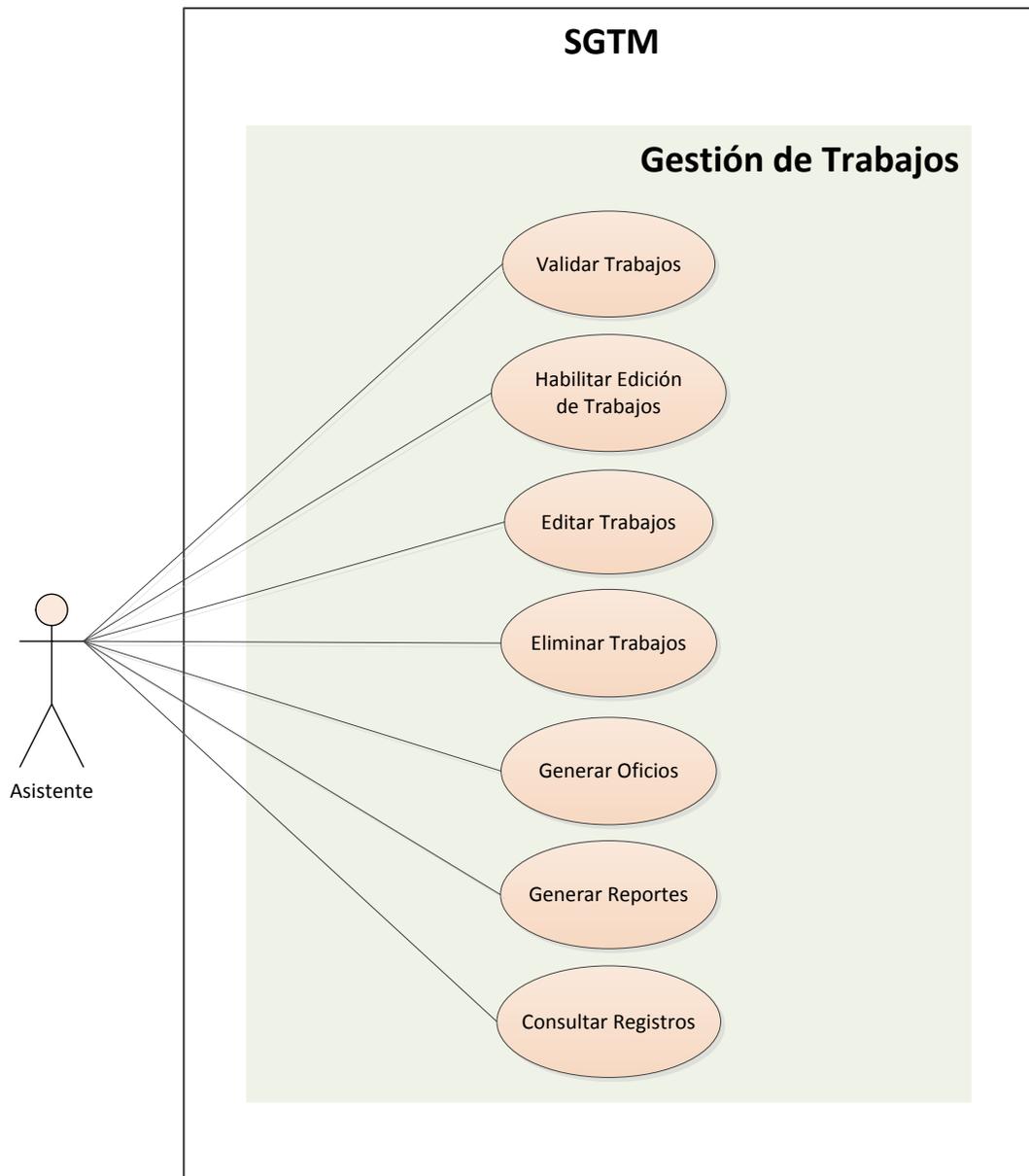


Figura 9. Caso de uso del actor Asistente (gestión de trabajos).

Diagrama de caso de uso del profesor

En la *Figura 10* se muestran las tareas o acciones que el actor profesor puede realizar, para el SGTM el usuario profesor, es el encargado de la gestión de los oficios de la academia.

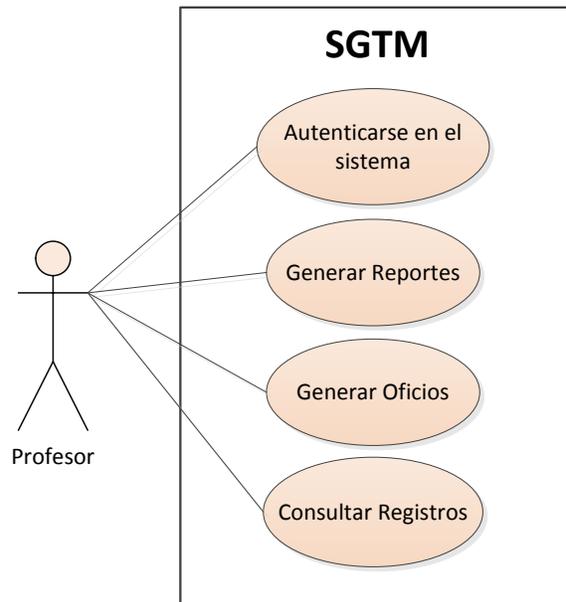


Figura 10. Caso de uso del actor Profesor.

Diagrama de caso de uso del alumno

En la *Figura 11* se muestran las tareas o acciones que el actor alumno puede realizar. Para el SGTM el usuario alumno, es el encargado del registro de los trabajos de titulación, así como de su gestión durante todo el proceso de titulación.

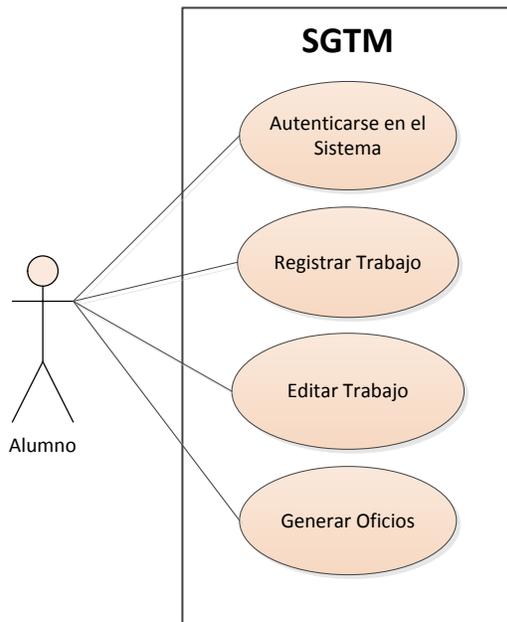


Figura 11. Caso de uso del actor Alumno.

Diagrama de caso de uso del administrador

En la *Figura 12* se muestran las tareas o acciones que el actor administrador puede realizar. Para el SGTM el usuario administrador, es el encargado de la gestión de usuarios sin importar la división a la que pertenezcan, así como la creación de nuevas divisiones, carreras y la gestión de los directores divisionales.

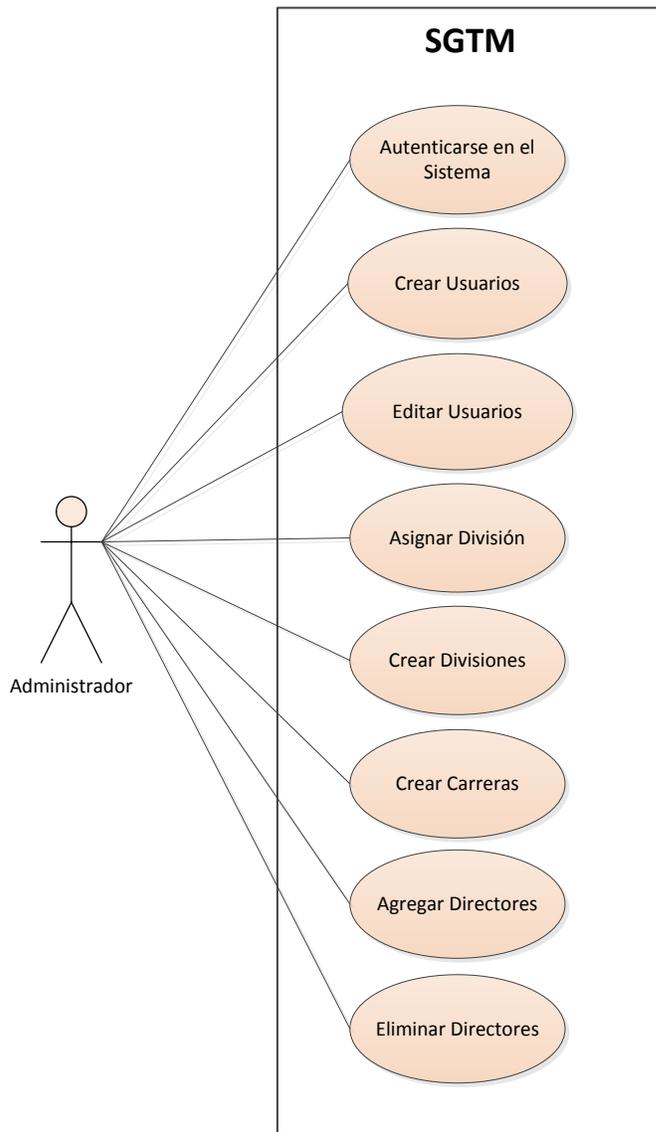


Figura 12. Caso de uso del actor Administrador.

4.2 Diseño

Mapa de navegación

Una parte importante del diseño de aplicaciones y sitios web, es el mapa de navegación.

El uso de los mapas de navegación es muy importante, ya que nos ayuda a tener una correcta distribución de las distintas secciones del sitio, de igual manera nos ayuda a agrupar todos aquellos apartados que tengan algún tipo de relación en común.

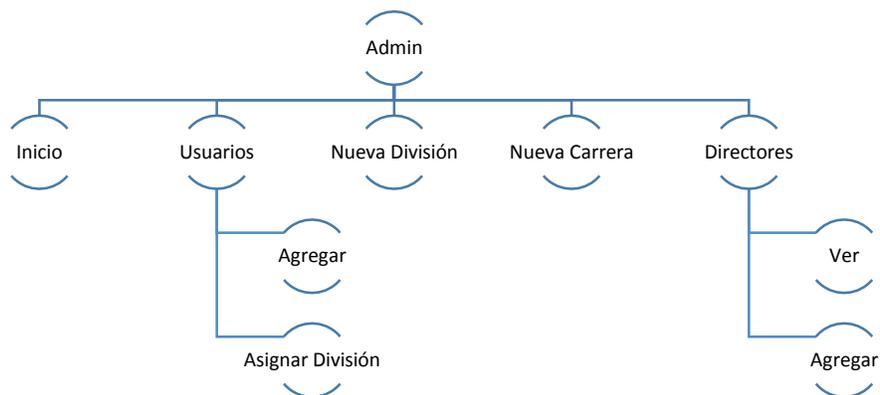


Figura 13. Mapa de navegación del administrador.

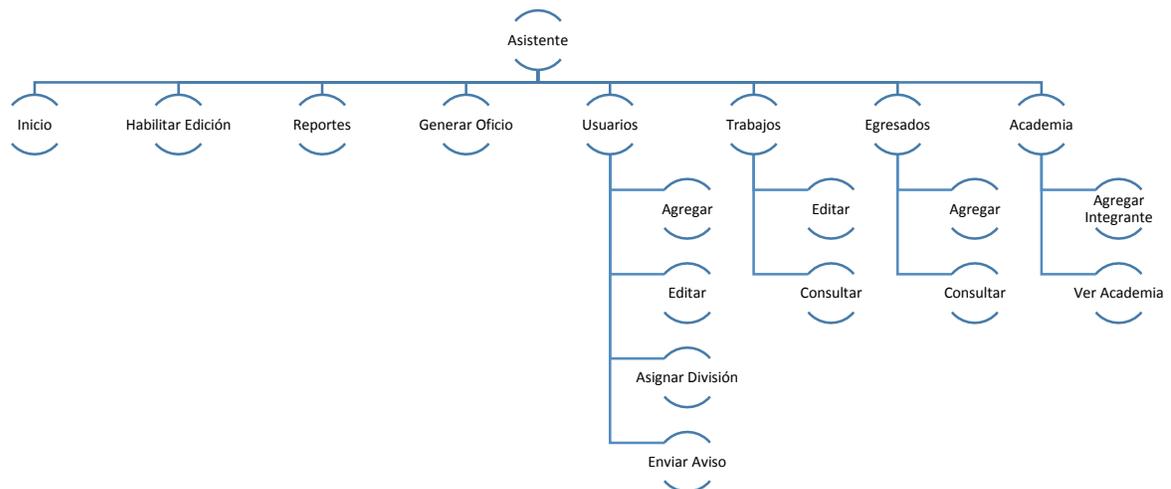


Figura 14. Mapa de navegación del/la asistente.

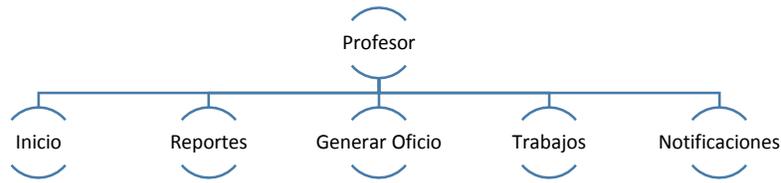


Figura 15. Mapa de navegación del profesor.

A diferencia de los demás usuarios, el mapa de navegación del alumno tendrá variaciones durante el registro de su trabajo.



Figura 16. Mapa de navegación del alumno sin trabajo registrado.

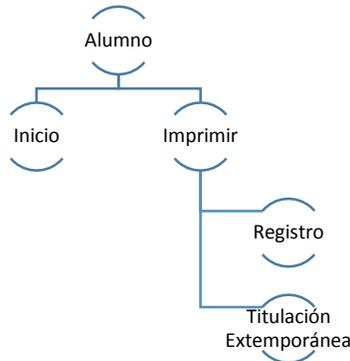


Figura 17. Mapa de navegación del alumno con trabajo pendiente.

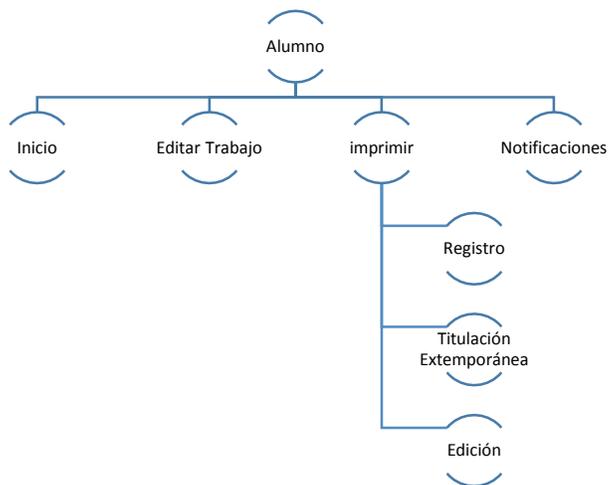


Figura 18. Mapa de navegación del alumno con trabajo aprobado.

La aplicación web está diseñada para una resolución de 1024 X 768 píxeles. Debido a que la aplicación perdería usabilidad si se accediera desde dispositivos móviles, se decidió utilizar medidas estáticas las cuales respetan los límites de seguridad del navegador.

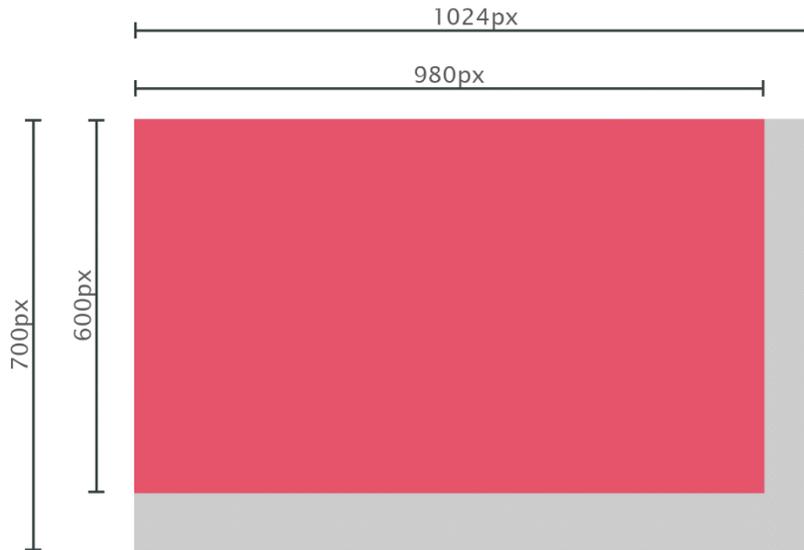


Figura 19. Resolución de pantalla.

Como se puede ver en la *Figura 19* se deja un espacio libre (área de seguridad), esto permite al contenido que exceda los 600 píxeles de alto pueda ser visualizado con la barra de navegación sin que esta interfiera con el contenido.

Estructura o *layout*

La estructura o *layout* define la ubicación de los espacios para la información que contendrá la aplicación. La estructura está diseñada para integrar toda la información en una resolución de 1024 X 768 píxeles respetando el espacio de seguridad.

En la *Figura 20* podemos observar los elementos que conforman la aplicación.

La información está distribuida en tres secciones principales:

- Encabezado: este contiene el logotipo de la institución, la interfaz de cierre de sesión y el título de la sección.
- Menú de navegación: contiene los menús y sub-menús de la aplicación.
- Contenido: contiene la información correspondiente a cada sección de la aplicación.

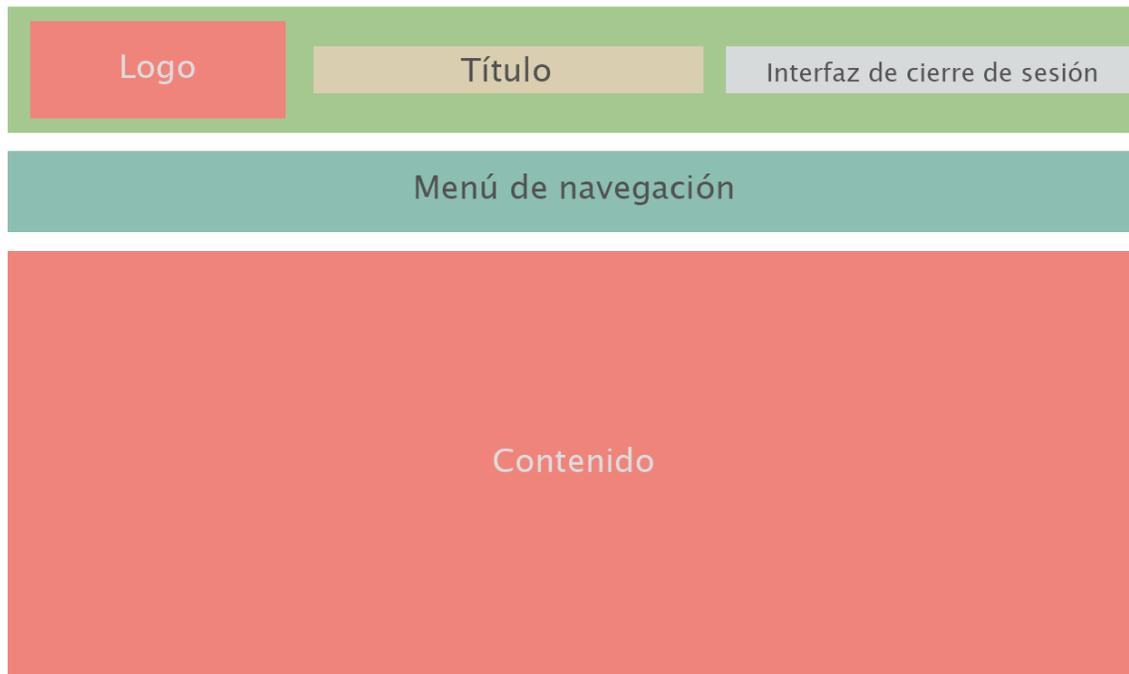


Figura 20. Diseño de la estructura de la aplicación.

Diseño de la base de datos

En esta etapa se construye el modelo E-R, por lo que se tuvo que analizar los tipos de relaciones que existen entre las distintas entidades.

En la *Figura 21* podemos observar que tenemos tres entidades centrales, las cuales permiten relacionar las demás entidades de la aplicación:

- Usuarios: se encarga de almacenar los datos correspondientes a cada usuario.
- División: se encarga de almacenar los datos referentes a las distintas divisiones.
- Carrera: se encarga de almacenar las carreras correspondientes a cada división.

De igual manera, se hacen uso de tablas pivote, esto para evitar el registro inadecuado de los trabajos de titulación, permitiendo al asistente validar o eliminar aquellos trabajos que así lo requieran.

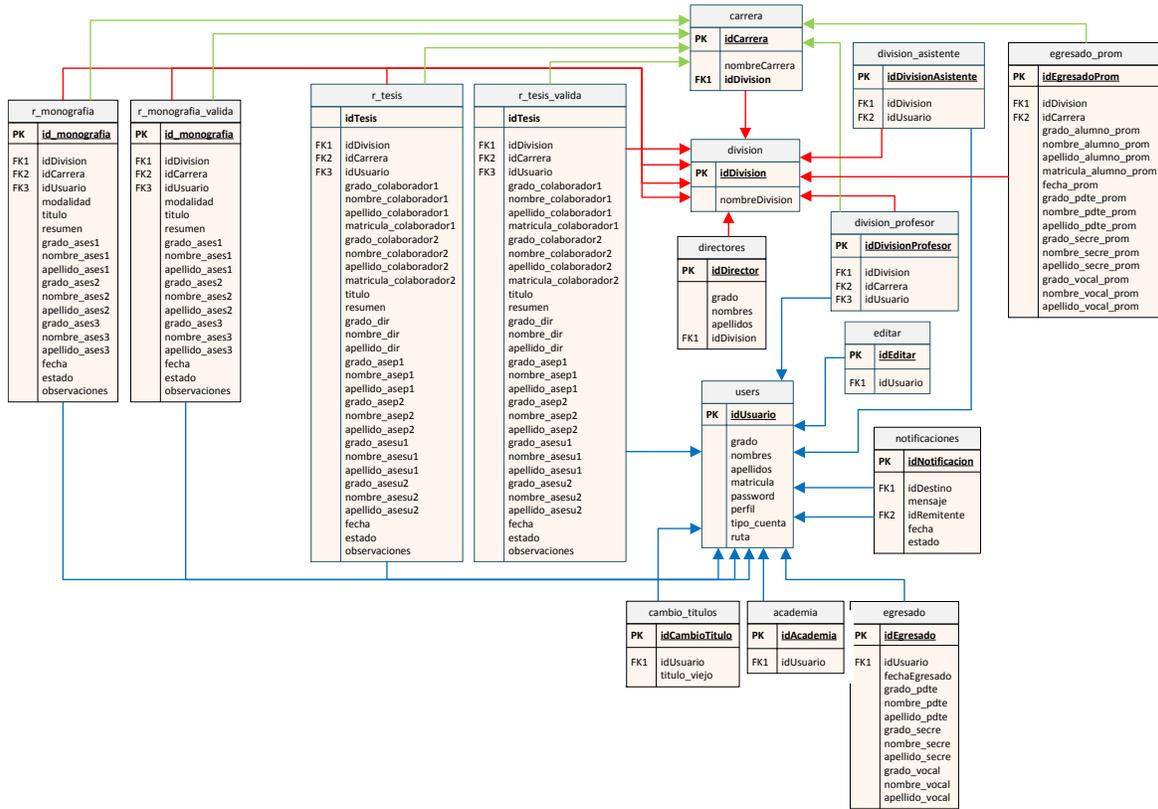


Figura 21. Diagrama E-R de la base de datos.

4.3 Codificación

Una vez que se tienen los requerimientos del sistema, el mapa de navegación y la estructura del sistema, se procede a realizar la codificación.

Para la codificación se utilizó el *Framework PHP CodeIgniter*; este *framework* trabaja bajo el patrón arquitectónico Modelo-Vista-Controlador, el cual nos permitirá separar la lógica del negocio de la interfaz gráfica de usuario.

En la *Figura 22* se puede observar cómo trabaja *CodeIgniter*.

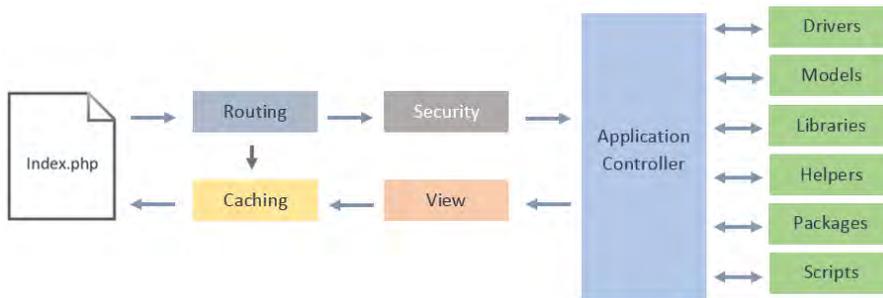


Figura 22. Diagrama de flujo de Codeigniter.

1. El `index.php` funciona como el controlador principal, este inicializa los recursos necesarios para correr *CodeIgniter*.
2. El routing examina la petición HTTP para determinar qué hacer con ella.
3. Si existe un archivo de caché, es enviado directamente al servidor.
4. Antes de que se cargue el controlador de la aplicación, las peticiones HTTP y cualquier dato enviado por el usuario es filtrado.
5. El controlador carga los modelos, las librerías del sistema, y cualquier otro recurso que sea necesario para procesar una petición.
6. Para finalizar, se prepara la vista para ser enviada al buscador, si está activado el cache, la vista se guarda en la cache, para ser utilizada en subsecuentes peticiones.

CodeIgniter trabaja con una estructura rígida de directorios, por lo que se deben seguir las reglas y colocar los scripts en el directorio donde corresponda. Inicialmente se tiene el directorio raíz de la aplicación *Figura 23*, este directorio se coloca de la manera habitual en el servidor.

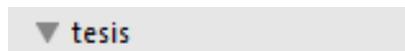


Figura 23. Directorio raíz de la aplicación.

Dentro del directorio raíz de la aplicación se encuentran los archivos de *CodeIgniter*

CodeIgniter cuenta con dos directorios principales:

- **System:** en este directorio se encuentran los scripts necesarios para que se pueda ejecutar *CodeIgniter*.
- **Application:** aquí es donde se guardarán los scripts de nuestra aplicación.

En el directorio *applications* es en el que nos enfocaremos, pues en este colocaremos los scripts de nuestra aplicación que se vayan construyendo.



Figura 24. Directorio application.

Inicialmente se modificaron algunos archivos del directorio config. En este directorios se encuentran los scripts de configuración de la aplicación.

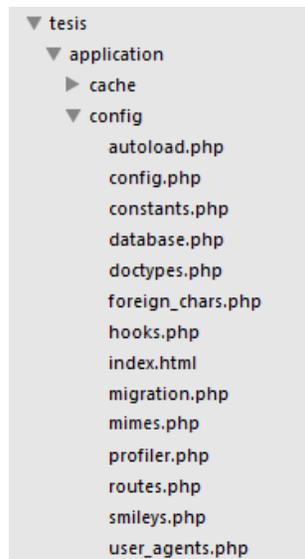


Figura 25. Directorio config.

Configuración de la aplicación

Inicialmente se configuró el archivo config.php, en este archivo se encuentra la base url de la aplicación, el juego de caracteres, la configuración de las cookies de sesión y el filtrado de código malicioso; en la tabla se pueden observar las configuraciones realizadas en la aplicación.

```
//Se configura la base url de la aplicación.
$config['base_url'] = 'http://formularios.uqroo.mx/tesis';
//Se configura la set de caracteres.
$config['charset'] = 'utf-8';
//Configuramos la llave de cifrado para las sesiones de CodeIgniter.
$config['encryption_key'] = 'aquí se colocó la llave de cifrado';
//Se configuran las sesiones.
$config['sess_cookie_name'] = 'ci_session';
$config['sess_expiration'] = 7200;
$config['sess_expire_on_close'] = TRUE;
//De esta forma nuestras cookies de sesión estarán cifradas y serán más seguras.
$config['sess_encrypt_cookie'] = TRUE;
$config['sess_use_database'] = FALSE;
$config['sess_table_name'] = 'ci_sessions';
$config['sess_match_ip'] = FALSE;
$config['sess_match_useragent'] = TRUE;
$config['sess_time_to_update'] = 300;
//Se permite el filtrado global de código malicioso.
$config['global_xss_filtering'] = TRUE;
```

Tabla 24. Configuración de config.php.

Posteriormente se configuró el archivo routes.php, en este archivo se coloca el controlador que se iniciará por default en la aplicación y urls amigables para acceder a las funciones de los controladores.

```
//Controlador que se iniciará por default.
$route['default_controller'] = 'login';
```

Tabla 25. Configuración de routes.php (default controller).

En este archivo también podemos crear las urls amigables, permitiendo acceder de una manera más legible a las funciones de los controladores.

```
//Rutas a las funciones con urls amigables.
$route['asistente/Habilitar-Edicion-De-Trabajo'] = "asistente/busca_editar";
```

Tabla 26. Configuración de routes.php (uri routing).

Lo que esto nos indica, es que cuando se acceda a “asistente/Habilitar-Edicion-De-Trabajo”, será re-direccionado a la función buscar_editar del controlador asistente.

Y finalmente configuramos el archivo database.php, en este archivo configuramos la conexión a la base de datos.

```
//Aquí se deja la opción por default ya que únicamente se ocupa una base de
datos
$active_group = 'default';
$active_record = TRUE;
//Configuración de la base de datos.
$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'Usuario de la base de datos';
$db['default']['password'] = 'Password del usuario de la base de datos';
$db['default']['database'] = 'tesis';
$db['default']['dbdriver'] = 'mysql';
$db['default']['dbprefix'] = '';
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = '';
$db['default']['char_set'] = 'utf8';
$db['default']['dbcollat'] = 'utf8_unicode_ci';
$db['default']['swap_pre'] = '';
$db['default']['autoinit'] = TRUE;
$db['default']['stricton'] = FALSE;
```

Tabla 27. Configuración de database.php.

Modelo – Vista – Controlador

CodeIgniter trabaja con el patrón MVC, la parte visual (interfaz de usuario) de la aplicación se coloca en `application/views`, en este directorio se encuentra todo el código HTML5 que permite crear el contenido visual de la aplicación; en el directorio `application/models` se encuentran las consultas SQL y en `application/controllers` los controladores o clases encargadas de conectar la parte visual con la lógica del negocio.

Modelo encargado del módulo de inicio de sesión

Los modelos de nuestra aplicación, se encuentran en `application/models`. Para crear un modelo encargado de la consulta a la base de datos para el ingreso de los usuarios a la aplicación, creamos un nuevo archivo dentro de `application/models/login` y se coloca el código que se muestra a continuación en la *Tabla 28*

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Login_model extends CI_Model {
    public function __construct() {
        parent::__construct();
        $this->load->database();
    } //End construct.
    public function login_user($usuario, $password){
        $this->db->select('*');
        $this->db->from('users');
        $this->db->where('usuario', $usuario);
        $this->db->where('password', $password);
        $query = $this->db->get();
        if ($query->num_rows() == 1) {
            return $query->result();
        } else{
            $this->session->
            >set_flashdata('usuario_incorrecto','Los datos introducidos son incorrectos
            verificalos');
            redirect(base_url().'login','refresh');
        }
    } //End login_user.
    public function loginUser365($usuario){
        $this->db->select('*');
        $this->db->from('users');
        $this->db->where('usuario', $usuario);
        $query = $this->db->get();
        if ($query->num_rows() == 1) {
            return $query->result();
        } else{
            redirect(base_url().'login','refresh');
        }
    }
} //End Logging_model.

```

Tabla 28. Modelo del asistente.

La función `login_user` selecciona todos los campos de la tabla `users` donde el nombre de usuario y `password` sean iguales a los que el usuario ha ingresado.

La función `loginUser365` selecciona todos los campos de la tabla `users` donde el nombre de usuario sea igual al usuario que se ha ingresado; esta función solo se utiliza cuando el usuario tiene una cuenta institucional activa de Office 365.

Para cargar al modelo, es necesario llamarlo en el controlador login utilizando el siguiente fragmento de código.

```
$this->load->model('login/login_model');
```

Tabla 29. Cargar el modelo.

Si se va a utilizar el modelo repetidas veces en el controlador, se recomienda cargarla en el constructor del controlador y sólo se hace la referencia a él de la siguiente manera.

```
$this->login_model->login_user ($usuario, $password);
```

Tabla 30. Referencia a una función del modelo.

De esta manera decimos que se va a utilizar la función login_user del modelo login_model y que además se le envían los parámetros \$usuario y \$password.

Para más detalles de este y otros modelos, véase [Anexo B](#).

Vista del módulo de inicio de sesión

Las vistas de nuestra aplicación, se encuentran en application/views. Para crear la vista encargada de mostrar la interfaz gráfica del inicio de sesión, creamos un nuevo archivo en application/views/login con el código que se encuentra en la *Tabla 31*.

```
<body>
<section class="contenedor">
    <div class="form_autentica">
        <?=form_open(base_url().'login/new_user')?>
            <label for="usuario">Usuario:</label>
            <input type="text" name="usuario" class="usuario"
placeholder="Usuario" value="<?= set_value('usuario');?>" /><span
class="cor_uqroo login">@uqroo.mx</span><p><?=form_error('usuario')?></p>
            <label for="password">Password:</label>
            <input type="password" name="password" class="pass"
placeholder="*****" /><p><?=form_error('password')?></p>
            <input type="hidden" name="token" value="<?= $token;?>">
            <input type="submit" value="Iniciar Sesión" class="btn_reg">
        <?=form_close()?>
    <?php
        if($this->session->flashdata('usuario_incorrecto')){
            ?>
            <p class="datos_incorrectos"><?=$this->session-
>flashdata('usuario_incorrecto')?></p>
            <?php
                }
            ?>
        </div>
```

```

<div class="office_365">
  <a href="<?= base_url();?>login/office365"></a>
</div>
</section><!--Contenedor-->

```

Tabla 31. Vista del módulo de inicio de sesión.

Como se observa en la tabla, hace falta contenido de la estructura básica de un documento HTML5, *CodeIgniter* permite dividir el código con lo cual se pueden reutilizar partes de ese código en distintas secciones de la aplicación.

Para cargar la vista, es necesario llamarla en el controlador o dentro de una misma vista utilizando el siguiente fragmento de código.

```
$this->load->view('login/login_view');
```

Tabla 32. Cargar la vista del módulo de sesión.

Para más detalles de este y otras vistas, véase [Anexo B](#).

Controlador del módulo de inicio de sesión

Los controladores de nuestra aplicación, se encuentran en `application/controllers`. Para crear el controlador que se encargue del inicio de sesión, creamos un nuevo archivo dentro de `application/controllers` y se coloca el código de la *Tabla 33*.

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Login extends CI_Controller {
    public function __construct(){
        parent::__construct();
        $this->load->model('login/login_model');
        $this->load-
>library(array('session','form_validation'));
        $this->load->library('office365');
        $this->load->model('global/global_model');
        $this->load->helper('url','form');
        $this->load->database('default');

    }//End __construct.
    public function index(){
        //Título del head y header.
        $titulo['titulo'] = "Iniciar Sesión";
        $titulo2['titulo'] = "Sistema de Gestión de Tesis y
Monografías";

        $this->load->view('global/head');
        $this->load->view('login/head_login',$titulo);
        $this->load->view('login/header_login',$titulo2);

```

```
switch ($this->session->userdata('perfil')) {
    case '':
        $data['token'] = $this->token();
        $this->load->view('login/login_view', $data);
    break;
    case 'administrador':
        redirect(base_url(). 'admin');
    case 'asistente':
        redirect(base_url(). 'asistente');
    case 'alumno':
        redirect(base_url(). 'alumno');
    case 'profesor':
        redirect(base_url(). 'profesor');
    default:
        $this->load->view('login/login_view');
    break;
} //End switch.
//Se carga el footer.
$this->load->view('global/footer');
} //End index.
} //End login.
```

Tabla 33. Controlador login.

El controlador login une la parte visual de la aplicación (vistas) con la lógica del negocio (modelos) del módulo de inicio de sesión, permitiendo verificar que tipo de usuario entra a la aplicación y de esta manera re-direccionarlo a su controlador respectivamente.

Para más detalles de este y otros controladores, véase [Anexo B](#).

Directorios adicionales

Adicionalmente a los directorios antes mencionados, se encuentran otros que nos pueden ser de ayuda en la creación de nuestras aplicaciones, sin embargo en este caso solo se utilizaron los siguientes:

- *Libraries*: en este directorio se crearon clases que nos ayudaron a tener una mejor organización del código y con esto evitamos crear repetidamente funciones en los controladores.
- *Helpers*: nos ayudaron a crear funciones sencillas que nos son de fácil acceso tanto en el controlador como en las vistas.
- *Language*: este directorio fue utilizado para crear mensajes en español para la verificación de los formularios, esta opción se puede omitir si se desean mostrar los errores de validación en inglés.

- **Errors:** se creó la página personalizada de error 404.

Para cargar las librerías y los helpers, utilizamos el mismo código de las vistas y los modelos.

```
//Cargamos una librería.  
$this->load->library('ruta de la librería').  
//Cargamos un helper.  
$this->load->helper('ruta del helper').
```

Tabla 34. Cargar una librería o *helper*.

Se pueden acceder a las librerías y los *helpers* en el controlador o en una vista siempre y cuando se haya cargado en el controlador donde se ejecute dicha vista.

Para acceder a una librería utilizamos el siguiente código.

```
$this->nombre de la librería->función de la librería();
```

Tabla 35. Uso de las funciones de una librería.

Para acceder a un helper utilizamos el siguiente código.

```
función del helper();
```

Tabla 36. Uso de las funciones de un helper.

4.4 Pruebas

Para la realización de las pruebas, se insertaron datos referentes a las carreras de nivel licenciatura con las que cuenta actualmente la División de Ciencias e Ingeniería.

Solicitud de registro de proyecto de titulación

Para registrar un trabajo ya sea de tesis o monografía, el alumno debe ingresar al sistema con su usuario y contraseña (cuenta manual) o con su cuenta de office 365 posteriormente ingresar los datos que se solicitan, para más detalles de esta acción véase [Anexo A \(registro del trabajo\)](#)

Registro en el sistema

UNIVERSIDAD DE QUINTANA ROO Registrar Trabajo 0910281@uqroo.mx

Inicio

Registro de Tesis

Ingresa todos los campos que se piden a continuación

Datos de la Carrera

*División: DIVISION DE CIENCIAS E INGENIERIA

*Carrera: INGENIERIA AMBIENTAL

Datos del Alumno

Nombre Completo: BR. STACEY LISELY ALPUCHE

Matrícula: 09-10281

Colaborador 1: Grado, Nombres, Apellidos

Matrícula: XX-XXXXX

Colaborador 2: Grado, Nombres, Apellidos

Matrícula: XX-XXXXX

Datos del Trabajo

*Título del Trabajo: CARACTERIZACIÓN FISICO

Resumen del Trabajo

Máximo 1000 caracteres.

1000

Datos del Director de Tesis			
*Nombre del Director	DR	VÍCTOR HUGO	DELGADO BLAS
Datos de los Asesores			
*Asesor Propietario 1	M.C	JOSÉ MARTÍN	RIVERO RODRÍGUEZ
*Asesor Propietario 2	DR	ADRIÁN	CERVANTES MARTÍNEZ
*Asesor Suplente 1	Q.F.B	JOSÉ LUIS	GONZÁLEZ BUCIO
*Asesor Suplente 2	DRA	MARTHA	GUTIÉRREZ ACURRE
Selecciona la Fecha Actual	01 /29/2014		
<input type="button" value="Enviar"/>			
<small>Todos los campos marcados con un * son obligatorios</small>			

Figura 26. Solicitud de registro de proyecto de titulación.

UNIVERSIDAD DE QUINTANA ROO Registrar Trabajo 0406098@uqroo.mx

Inicio

Registro de Monografía

Ingresa todos los campos que se piden a continuación

Datos de la Carrera

*División: DIVISIÓN DE CIENCIAS E INGENIERÍA

*Carrera: INGENIERÍA EN REDES

Datos del Alumno

Nombre Completo: BR. FEDERICO OSORIO ESCALANTE

Matrícula: 04-06098

Datos del Trabajo

*Modalidad: INVESTIGACION DOCUMENTAL

*Título del Trabajo: SEGURIDAD PERIMETRAL

Resumen del Trabajo

Máximo 1000 caracteres.

1000

Datos del Comité de Supervisión

*Asesor 1: M.S.I. LAURA YÉSICA DÁVALOS CASTILLA

*Asesor 2: M.T.I. VLADIMIR VENIAMIN CABAÑAS VICTORIA

*Asesor 3: M.S.I. RUBÉN ENRIQUE GONZÁLEZ ELIXAVIDE

Selecciona la Fecha Actual: 09/24/2014

Enviar

Todos los campos marcados con un * son obligatorios

Figura 27. Registro de una Monografía.

Impresión de oficio de registro

Posteriormente a la solicitud del registro del trabajo, se procede a imprimir el oficio de solicitud de registro el cual deberá ser entregado a la asistente junto con el oficio de la academia, véase [impresión de oficio de la academia](#); para imprimir el oficio de registro, es necesario ingresar los datos solicitados, para más detalles de esta acción véase [Anexo A Generar oficio](#).

Cd. Chetumal, Quintana Roo; a 29 de Enero del 2014

Dr. José Hernández Rodríguez
Presidente del Consejo Divisional
División de Ciencias e Ingeniería
PRESENTE

Por este medio me permito solicitarle se someta a la aprobación del Consejo Divisional de Ciencias e Ingeniería el proyecto de tesis con el título "Caracterización Físico-Química Y Del Zooplancton En La Laguna La Sabana, Quintana Roo.", como requisito para darle seguimiento al trámite de registro de acuerdo con el Artículo 101 B del R.E.T.L.

El proyecto de tesis estará dirigido por Dr. Víctor Hugo Delgado Blas.

Asimismo, de acuerdo al Artículo 105 del R.E.T.L. se propone la integración del Comité de Tesis como sigue:

Asesores propietarios

M.C. José Martín Rivero Rodríguez
Dr. Adrián Cervantes Martínez

Asesores Suplentes

Q.F.B. José Luis González Bucio
Dra. Martha Gutiérrez Aguirre

Atentamente

Br. Stacey Lisely Alpuche
Matrícula: 09-10281

Visto Bueno

DR. Víctor Hugo Delgado Blas
Director (a) del proyecto de tesis

Figura 28. Oficio de registro de una Tesis.

Cd. Chetumal, Quintana Roo; a 24 de Septiembre del 2014

Dr. José Hernández Rodríguez
Director de la División de Ciencias e Ingeniería
PRESENTE

Por este medio me permito solicitarle su aprobación de la propuesta de trabajo monográfico en la modalidad Investigación Documental, con el título "Seguridad Perimetral", asimismo, se propone la integración del Comité de Supervisión, los cuales firman de visto bueno con respecto al trabajo monográfico, como requisito para darle seguimiento al trámite de registro de acuerdo al Artículo 100 G del R.E.T.L.

Comité de Supervisión:

M.S.I. Laura Yésica Dávalos Castilla
Asesor

M.T.I. Vladimir Veniamin Cabañas Victoria
Asesor

M.S.I. Rubén Enrique González Elixavide
Asesor

Atentamente

Br. Federico Osorio Escalante
Matrícula: 04-06098

C.c.p Interesado

Figura 29. Oficio de registro de una Monografía.

Impresión de oficio del Área Académica

Posteriormente al registro del trabajo, se procede a imprimir el oficio de la academia el cual deberá ser entregado al alumno para que junto con el oficio de registro sea entregado a la asistente, para imprimir el oficio de la academia, es necesario ingresar los datos solicitados por la aplicación, para más detalles de esta acción véase [Anexo A generar oficio](#).

UNIVERSIDAD DE QUINTANA ROO Oficio Academia victordel@uqroo.mx

Inicio Reportes Oficios Trabajos Notificaciones

Oficio Academia

Ingresa la matrícula del alumno (sin guión) y la fecha.

*Alumno @uqroo.mx

*Asunto

*Fecha Actual

Todos los campos marcados con un * son obligatorios

Figura 30. Generar Oficio del Área Académica.



UNIVERSIDAD DE QUINTANA ROO

DIVISIÓN DE CIENCIAS E INGENIERÍA

Cd. Chetumal, Quintana Roo; a 29 de Enero del 2014

Br. Stacey Lisely Alpuche

El Área de Ingeniería Ambiental revisó el proyecto de Tesis denominado "Caracterización Físico-Química Y Del Zooplancton En La Laguna La Sabana, Quintana Roo.", aprobando el tema y dando Visto Bueno al Comité de Tesis conformado por Dr. Víctor Hugo Delgado Blas como Director, M.C. José Martín Rivero Rodríguez y Dr. Adrián Cervantes Martínez como Asesores Propietarios y Q.F.B. José Luis González Bucio y Dra. Martha Gutiérrez Aguirre como Asesores Suplentes.

Atentamente

"Fructificar la Razón: Trascender Nuestra Cultura"

Integrantes del Área

Dr. José Alfonso Canche Uuh

Dr. José Manuel Carrión Jiménez

M.I. Norma Angélica Oropeza García

Dr. Víctor Hugo Delgado Blas

M.C. José Martín Rivero Rodríguez

Biól. Laura Patricia Flores Castillo

I.Q. José Luis Guevara Franco

C.e.p. _ Jefe Depto._

Boulevard Bahía s/n, esquina Ignacio Comonfort, Colonia del Bosque, Código Postal 77019, Chetumal, Quintana Roo, México. Teléfono/Fax
+(983)83.50300. www.uqroo.mx

Figura 31. Oficio de la academia (tesis).



UNIVERSIDAD DE QUINTANA ROO

DIVISIÓN DE CIENCIAS E INGENIERÍA

Cd. Chetumal, Quintana Roo; a 24 de Septiembre del 2014

Br. Federico Osorio Escalante

El Área de Ingeniería en Redes revisó la propuesta del trabajo monográfico en la modalidad Investigación Documental denominado "Seguridad Perimetral", aprobándolo y dando Visto Bueno al Comité de Supervisión conformado por M.S.I. Laura Yésica Dávalos Castilla, M.T.I. Vladimir Veniamin Cabañas Victoria y M.S.I. Rubén Enrique González Elixavide como asesores.

Atentamente

“Fructificar la Razón: Trascender Nuestra Cultura”

Integrantes del Área

M.T.I. Melissa Blanqueto Estrada

Dr. Freddy Ignacio Chan Puc

Dr. Homero Toral Cruz

Dr. Jaime Silverio Ortegón Aguilar

M.S.I. Laura Yésica Dávalos Castilla

Dr. Victor Manuel Sánchez Huerta

M.T.I. Vladimir Veniamin Cabañas Victoria

M.C. Javier Vázquez Castillo

M.S.I. Rubén Enrique González Elixavide

C.c.p. _ Jefe Depto._

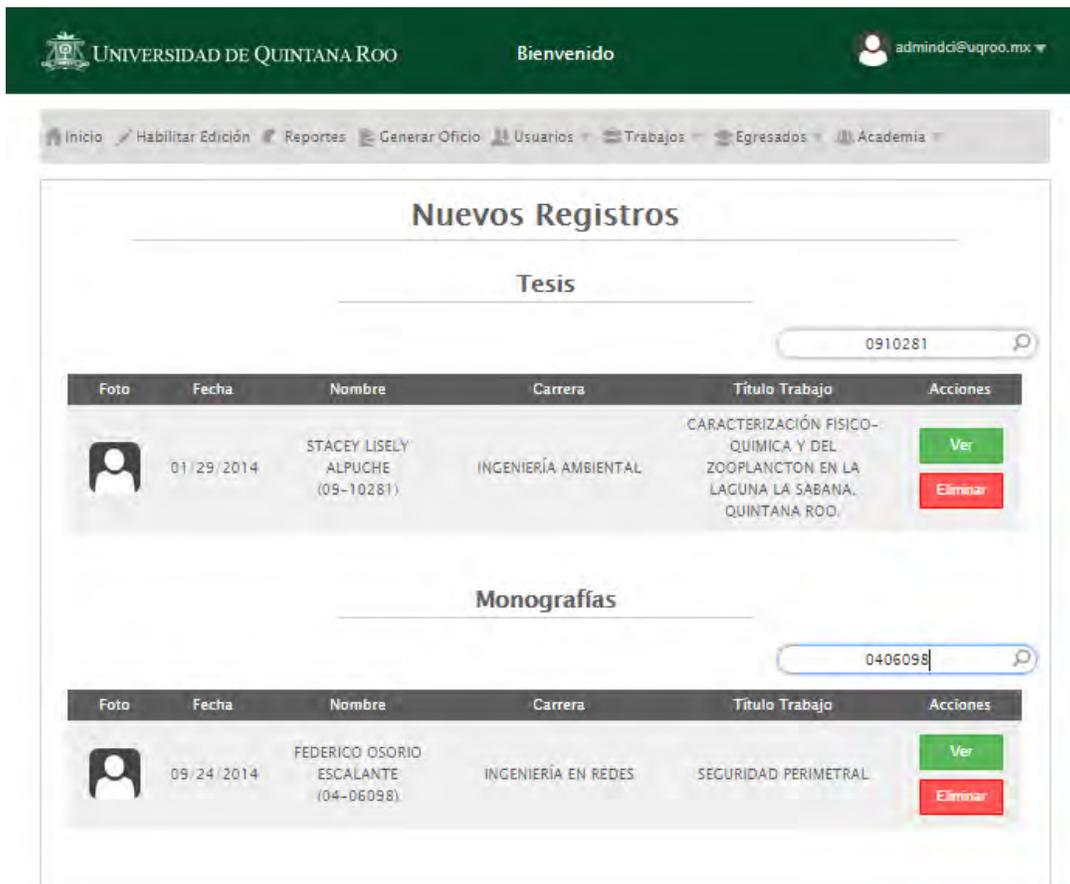
Boulevard Bahía s/n, esquina Ignacio Comonfort, Colonia del Bosque, Código Postal 77019, Chetumal, Quintana Roo, México. Teléfono/Fax
+(983)83.50300, www.uqroo.mx

Figura 32. Oficio de la academia (monografía).

Aprobación del trabajo e impresión del oficio

Una vez que el Consejo Divisional apruebe el trabajo registrado, se procede a darle el estatus de aprobado en el sistema, para más detalles de esta acción véase [Anexo A validar trabajo](#).

Aprobación de trabajos



The screenshot displays the 'Nuevos Registros' (New Records) interface. At the top, there is a navigation bar with the university logo and name, a welcome message, and a user profile. Below this is a menu bar with options like 'Inicio', 'Habilitar Edición', 'Reportes', 'Generar Oficio', 'Usuarios', 'Trabajos', 'Egresados', and 'Academia'. The main content area is divided into two sections: 'Tesis' and 'Monografías'. Each section has a search bar and a table of records. The 'Tesis' section shows a record for Stacey Lisely Alpuche, and the 'Monografías' section shows a record for Federico Osorio Escalante. Both records have 'Ver' (View) and 'Eliminar' (Delete) buttons.

Foto	Fecha	Nombre	Carrera	Título Trabajo	Acciones
	01/29/2014	STACEY LISELY ALPUCHE (05-10281)	INGENIERÍA AMBIENTAL	CARACTERIZACIÓN FÍSICO-QUÍMICA Y DEL ZOOPLANCTON EN LA LAGUNA LA SABANA, QUINTANA ROO.	Ver Eliminar

Foto	Fecha	Nombre	Carrera	Título Trabajo	Acciones
	09/24/2014	FEDERICO OSORIO ESCALANTE (04-06098)	INGENIERÍA EN REDES	SEGURIDAD PERIMETRAL	Ver Eliminar

Figura 33. Trabajo registrado (ventana principal del asistente).

Una vez que damos clic en el botón ver, se muestran los detalles del registro, para posteriormente dar click en *Aprobar*.

UNIVERSIDAD DE QUINTANA ROO **Detalle Registro** admin@uqroo.mx

Inicio / Habilitar Edición / Reportes / Generar Oficio / Usuarios / Trabajos / Egresados / Academia

Detalle del Registro

Datos de la Carrera
División: DCI
Carrera: IAM

Datos del Alumno
Nombre Alumno: BR. STACEY LISELY ALPUCHE
Matrícula: 09-10281

Datos de los Colaboradores
Nombre Colaborador 1: NO REGISTRADO
Nombre Colaborador 2: NO REGISTRADO

Datos del Trabajo
Título del Trabajo: CARACTERIZACIÓN FÍSICO-QUÍMICA Y DEL ZOOPLANCTON EN LA LAGUNA LA SABANA, QUINTANA ROO.
Resumen del Trabajo: NO REGISTRADO

Datos del Director
Director: DR. VÍCTOR HUGO DELGADO BLAS

Datos de los Asesores
Asesor Propietario 1: M.C. JOSÉ MARTÍN RIVERO RODRÍGUEZ
Asesor Propietario 2: DR. ADRIÁN CERVANTES MARTÍNEZ
Asesor Suplente 1: Q.F.B. JOSÉ LUIS GONZÁLEZ BUCIO
Asesor Suplente 2: DRA. MARTHA GUTIÉRREZ AGUIRRE

Fecha de Registro
01/29/2014

Estado
PENDIENTE

Observaciones

[Aprobar](#)

Figura 34. Detalles del registro de un trabajo de tesis.

UNIVERSIDAD DE QUINTANA ROO **Detalle Registro** adminhci@uqroo.mx

Inicio / Habilitar Edición / Reportes / Generar Oficio / Usuarios / Trabajos / Egresados / Academia

Detalle del Registro

Datos de la Carrera
División: DCI
Carrera: IR

Datos del Alumno
Nombre Alumno: BR. FEDERICO OSORIO ESCALANTE
Matrícula: 04-06098

Datos del Trabajo
Modalidad: INVESTIGACIÓN DOCUMENTAL
Título del Trabajo: SEGURIDAD PERIMETRAL
Resumen del Trabajo: NO REGISTRADO

Datos de los Asesores
Asesor 1: M.S.I. LAURA YÉSSICA DÁVALOS CASTILLA
Asesor 2: M.T.J. VLADIMIR VENIAMIN CABAÑAS VICTORIA
Asesor 3: M.S.I. RUBÉN ENRIQUE GONZÁLEZ ELIXAVIDE

Fecha de Registro
09/24/2014

Estado
PENDIENTE

Observaciones

[Aprobar](#)

Figura 35. Detalle del registro de un trabajo monográfico.

Impresión de oficio de aprobación

Posteriormente a la modificación del estatus de aprobado, se procede a imprimir el oficio de aprobación; para imprimir el oficio de aprobación, es necesario ingresar los datos solicitados por la aplicación, para más detalles de esta acción véase [Anexo A generar oficio](#).



Consejo Divisional

Cd. Chetumal, Quintana Roo; a 03 de Febrero del 2014

UQROO/DCI/CD/142/14

"2014, Año de Octavio Paz"

Br. Stacey Lisely Alpuche

Matrícula 09-10281

Presente

Se le notifica que el Consejo Divisional, en su sesión ordinaria del **03 de Febrero del 2014**, acordó aprobar su solicitud de registro de tesis titulado: "**Caracterización Físico-Química Y Del Zooplancton En La Laguna La Sabana, Quintana Roo.**". Asimismo, se aprueba el Comité de Tesis propuesto, quedando integrado como se indica:

Director: Dr. Víctor Hugo Delgado Blas

Asesores Propietarios:

M.C. José Martín Rivero Rodríguez

Dr. Adrián Cervantes Martínez

Asesores Suplentes:

Q.F.B. José Luis González Bucio

Dra. Martha Gutiérrez Aguirre

Por su atención a la presente, quedo de usted.

Atentamente

"Fructificar la razón: trascender nuestra cultura"

Dr. José Hernández Rodríguez

Presidente

C.c.p.- Comité de Tesis.

C.c.p.- Archivo.

JHR/immib*

Boulevard Bahía s/n, esquina Ignacio Comonfort, Colonia del Bosque, Código Postal 77019, Chetumal, Quintana Roo, México. Teléfono/Fax
+(983)83.50300, www.uqroo.mx

Versión: Enero 23, 2014

Código: REC-001/FO-001

Documento impreso o electrónico que no se consulte directamente en el portal SIGC (<http://sigc.uqroo.mx/>) se considera COPIA NO CONTROLADA

Figura 36. Oficio de respuesta a trabajo de tesis.



Cd. Chetumal, Quintana Roo; a 29 de Septiembre del 2014

UQROO/DCI/524/2014

"Año 2014, 40 aniversario del Estado Libre y Soberano de Quintana Roo"

Br. Federico Osorio Escalante
Ingeniería en Redes
presente

Con el presente se le notifica el registro de su trabajo monográfico en la modalidad de Investigación Documental, titulado: "**Seguridad Perimetral**", así como la autorización del Comité Supervisor integrado por los profesores, M.S.I. Laura Yésica Dávalos Castilla, M.T.I. Vladimir Veniamin Cabañas Victoria y M.S.I. Rubén Enrique González Elixavide.

Sin otro particular, le envío un cordial saludo.

Atentamente
"Fructificar la razón: trascender nuestra cultura"

Dr. José Hernández Rodríguez
Director

C.c.p.- Archivo.

Boulevard Bahía s/n, esquina Ignacio Comonfort, Colonia del Bosque, Código Postal 77019, Chetumal, Quintana Roo, México. Teléfono/Fax
+(983)83.50300, www.uqroo.mx

Versión: Enero 23, 2014

Código: REC-001/PO-001

Documento impreso o electrónico que no se consulte directamente en el portal SIGC (<http://sigc.uqroo.mx/>) se considera COPIA NO CONTROLADA

Figura 37. Oficio de respuesta a trabajo monográfico.

Usabilidad del sistema

Para las pruebas de usabilidad se le presentó el sistema al personal administrativo de la DCI debido a que en ellas se centran gran parte de las tareas que se pueden realizar en este.

Inicialmente se les mostró la manera en la que se registran los usuarios del sistema, para posteriormente poder validar los trabajos registrados y generar los oficios correspondientes.

Una vez que se realizaron las pruebas del proceso de registro y validación, se procedió a mostrarles las demás características del sistema, entre las cuales se encuentran las consultas de los trabajos y egresados, estadísticas divisionales y el registro de los integrantes de las distintas academias.

Durante el proceso se pudo observar que las asistentes se mostraban optimistas con el nuevo sistema y se vio una buena aceptación el cual se ve reflejado en la encuesta de usabilidad, véase [Anexo D](#).

Las pruebas con los demás usuarios, se realizarán posteriormente, sin embargo muchas de las tareas son semejantes entre los distintos usuarios.

CONCLUSIÓN

Conclusión

Con la realización de esta aplicación web se ha conseguido lograr la sistematización de un proceso administrativo que involucra a estudiantes y profesores. El resultado es una herramienta que tiene como propósito facilitar la gestión de los trabajos de titulación.

Para el desarrollo de la aplicación se propuso el uso del *Framework* PHP *CodeIgniter*, ya que trabaja bajo el patrón arquitectónico MVC; la principal ventaja del uso de este entorno de trabajo es la reutilización de código y alta escalabilidad en las aplicaciones.

Para la realización de esta aplicación web, se tuvieron que identificar los requerimientos del sistema, dentro de los requerimientos se definieron cuatro tipos de usuarios y cada uno de ellos con acciones específicas.

Después de haber definido los requerimientos, se realizó el diseño en el cual se definió el mapa de navegación para crear la representación esquemática de la aplicación, posteriormente se realizó el diseño de la base de datos procurando tener la mínima redundancia posible y finalmente se realizó el diseño de la estructura o layout definiendo cada espacio de información que contendría la aplicación.

Ya teniendo claro los requerimientos y la estructura de la aplicación, se procedió a realizar la codificación enfocándonos primordialmente en los controladores seguidamente de los modelos y finalmente de las vistas.

Ya que uno de los requerimientos es dar soporte a cuatro tipos de usuario, se realizaron cuatro controladores donde cada uno de ellos se encarga de proporcionarle al usuario las funcionalidades que se definieron en los requerimientos; aunado al desarrollo de los controladores se crearon los modelos encargados de la base de datos procurando agrupar las consultas en común entre los diferentes usuarios.

Finalmente se crearon las vistas que están encargadas de la representación gráfica de la información. Para las vistas se procuró tener un diseño que se adapte a la imagen institucional y brinde una correcta distribución de la información cumpliendo con las reglas de usabilidad de las aplicaciones web.

Durante la presentación del sistema a las asistentes, se pudo observar una buena aceptación de este debido a que no se les complicó el manejo y pudieron observar nuevas características que les ayudarán en su trabajo cotidiano.

Podemos concluir que los objetivos planteados fueron alcanzados, teniendo como principal aportación una aplicación web que ayudará a los usuarios a tener una gestión eficiente de los trabajos de titulación a nivel licenciatura.

TRABAJO FUTURO

Trabajo futuro

Como trabajo futuro se puede:

- Dar soporte para el registro de dos o más carreras simultáneamente.

El sistema actualmente no cuenta con soporte para el registro de dos carreras o más, sin embargo tiene la flexibilidad de realizar cambios sin comprometer el funcionamiento actual.

- Implementar el sistema en otras divisiones de la Universidad de Quintana Roo Campus Chetumal.

El sistema se encuentra listo para implementarse en otras divisiones de la Universidad de Quintana Roo, únicamente basta con realizar el debido registro de los usuarios en el sistema.

- Dar soporte al registro de trabajos de posgrado.

Una de las ventajas del desarrollo de aplicaciones bajo el patrón MVC y en específico con el *Framework PHP CodeIgniter*, es la escalabilidad de las aplicaciones, permitiendo incorporar nuevas funcionalidades que permitan dar soporte no solo a un grupo de usuarios, sino a todo aquel que lo requiera.

BIBLIOGRAFÍA

Bibliografía

1. Cano, David Moner. [En línea] [Citado el: 28 de 05 de 2015.] http://www.unsj.edu.ar/unsjVirtual/comunicacion/seminarionuevastecnologias/wp-content/uploads/2015/04/usabilidad_para_web.pdf.
2. Pitt, Chris. *Pro PHP MVC*. s.l. : Apress, 2012. 978-1-4302-4165-2.
3. Bahit, Eugenia. *POO y MVC en PHP*.
4. López, Alejandro Rivera. Universidad de las Américas Puebla. [En línea] 16 de 01 de 2008. [Citado el: 19 de 01 de 2015.] http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/rivera_l_a/capitulo2.pdf.
5. Zend *Framework*. [En línea] 2006. [Citado el: 21 de 01 de 2015.] <http://framework.zend.com/about/>.
6. CakePHP. [En línea] 2005. [Citado el: 21 de 01 de 2015.] http://book.cakephp.org/2.0/_downloads/en/CakePHPCookbook.pdf.
7. *CodeIgniter*. [En línea] 2006. [Citado el: 21 de 01 de 2015.] <http://www.CodeIgniter.com/>.
8. Gauchat, Juan Diego. *El gran libro de HTML5, CSS3 y JavaScript*. Barcelona : Marcombo, 2012. 978-84-267-1782-5.
9. Lunn, Ian. *CSS3 Foundations*. s.l. : John Wiley & Sons, 2013. 978-1-118-35654-8.
10. McFarland, David Sawyer. *JavaScript & jQuery: The Missing Manual*. Third. s.l. : O'Reilly, 2014. 978-1-491-94707-4.
11. Weinberg, James R. Groff & Paul N. *SQL Manual de referencia*. s.l. : McGraw-Hill, 2003. 84-481-3930-5.
12. Aguilar, Luis Joyanes. *Fundamentos de Programación*. Tercera. s.l. : Mc Graw Hill.
13. w3schools. [En línea] 2015. [Citado el: 20 de 01 de 2015.] http://www.w3schools.com/css/css3_intro.asp.
14. JQuery. [En línea] 2015. [Citado el: 22 de 01 de 2015.] <https://code.jquery.com/>.
15. Google. [En línea] 16 de 01 de 2015. [Citado el: 22 de 01 de 2015.] <https://developers.google.com/speed/libraries/>.

ANEXO A

Requerimientos funcionales

Los requerimientos funcionales son de mucha ayuda a la hora de analizar los requerimientos de un sistema, ya que en ellos describimos verbalmente las acciones que realiza cada usuario en él.

Actor: Alumno

Autenticarse en el sistema	
Función	Autenticarse en el sistema para poder hacer uso de él.
Descripción	Permite al alumno ingresar al sistema para hacer uso de su interfaz.
Entradas	Usuario, contraseña.
Salida	Interfaz de usuario.
Proceso	El alumno ingresan en el sistema su nombre de usuario y contraseña, el sistema validará si el usuario existe y si su contraseña es correcta. Si es así, el sistema mostrará al usuario la interfaz correspondiente a su perfil en el sistema.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 37. Especificación del requerimiento autenticarse en el sistema (alumno).

Actor: Alumno.

Registro del trabajo	
Función	Registrar los datos requeridos del trabajo en el sistema.
Descripción	Permite al alumno registrar en el sistema los datos de su trabajo de titulación.
Entradas	Datos del trabajo: fecha de registro, título del trabajo, nombre del director de tesis, nombre de los asesores propietarios(tesis), nombre de los asesores suplentes(tesis), modalidad(monografía), comité de supervisión(monografía), resumen.
Salida	Confirmación de datos ingresados.
Proceso	El sistema desplegará el formulario para el registro de la tesis o monografía, en el cual se deberán de ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos requeridos se validarán los datos y se registra el trabajo.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 38. Especificación del requerimiento registro del trabajo.

Actor: Alumno

Generar oficios	
Función	Generar el oficio de registro o edición del trabajo de titulación.
Descripción	Permite al alumno generar su oficio de registro o edición de su trabajo de titulación.
Entradas	Fecha actual y tipo de oficio.
Salida	Oficio del registro o edición del trabajo en formato PDF.
Proceso	Una vez registrado el trabajo o realizada alguna edición, se deberá seleccionar el tipo de oficio a imprimir e ingresar la fecha actual, posteriormente se mostrara en la pantalla el oficio en formato PDF para poder imprimirlo.
Restricciones	

Tabla 39. Especificación del requerimiento generar oficios (alumno).

Actor: Alumno

Editar trabajo	
Función	Editar los datos del trabajo.
Descripción	Permite al alumno editar los datos del trabajo, en dado caso que exista algún cambio en su título o sus asesores (si es tesis) o en su comité de supervisión (si es monografía).
Entradas	Datos del trabajo: título del trabajo, comité de tesis o comité de supervisión.
Salida	Confirmación de datos ingresados.
Proceso	El sistema desplegará el formulario para la edición de la tesis o monografía, en el cual se deberán editar los datos -descritos en la entrada-. Después de ser editados los datos se validarán los datos y se registra el trabajo.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente. Para realizar modificaciones en los trabajos, el alumno deberá acudir con la asistente para que active la edición de su trabajo, de lo contrario no podrá realizar ninguna modificación.

Tabla 40. Especificación del requerimiento editar trabajo (alumno).

Actor: Asistente

Autenticarse en el sistema	
Función	Autenticarse en el sistema para poder hacer uso de él.
Descripción	Permite al asistente ingresar al sistema para hacer uso de su interfaz.
Entradas	Usuario, contraseña.
Salida	Interfaz de usuario.
Proceso	El asistente ingresa en el sistema su nombre de usuario y contraseña, el sistema validará si el usuario existe y si su contraseña es correcta. Si es así, el sistema mostrará al usuario la interfaz correspondiente a su perfil en el sistema.
Restricciones	Si el nombre de usuario y contraseña no son correctos, el sistema mostrará un mensaje informando que los datos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 41. Especificación del requerimiento autenticarse en el sistema (asistente).

Actor: Asistente

Crear usuarios	
Función	Crear nuevos usuarios del sistema.
Descripción	Permite al asistente crear nuevos usuarios del sistema.
Entradas	Grado, nombres, apellidos, nombre de usuario, contraseña, perfil del usuario, tipo de cuenta.
Salida	Mensaje de confirmación.
Proceso	El asistente deberá seleccionar la opción "usuarios", en seguida "agregar" e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el usuario.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 42. Especificación del requerimiento crear usuarios.

Actor: Asistente

Editar usuarios	
Función	Editar usuarios del sistema.
Descripción	Permite al asistente editar a los usuarios del sistema.
Entradas	Grado, nombres, apellidos, nombre de usuario, contraseña, perfil del usuario, tipo de cuenta.
Salida	Mensaje de confirmación.
Proceso	El asistente deberá seleccionar la opción “usuarios”, en seguida “editar” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el usuario.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 43. Especificación del requerimiento editar usuarios.

Actor: Asistente

Asignar división	
Función	Asignar división a los usuarios.
Descripción	Permite al asistente asignarles una división a los usuarios con perfil de administrador y profesor.
Entradas	Nombre de usuario y carrera (en caso de ser un profesor).
Salida	Mensaje de confirmación.
Proceso	El asistente deberá seleccionar la opción “usuarios”, en seguida “asignar división” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el usuario.
Restricciones	La división que se agregará por defecto a los usuarios es a la que pertenece la asistente.

Tabla 44. Especificación del requerimiento asignar división (asistente).

Actor: Asistente

Validar trabajo	
Función	Validar los trabajos registrado por el alumno.
Descripción	Permite al asistente marcar como aprobados los trabajos registrados por los alumnos.
Entradas	Observaciones (si son requeridas).
Salida	Mensaje de confirmación.
Proceso	El asistente deberá buscar el trabajo y dar click en aprobar.
Restricciones	Solo deberá cambiar el estado de los trabajos que han sido aprobados por el consejo divisional.

Tabla 45. Especificación del requerimiento valida trabajo.

Actor: Asistente

Eliminar trabajos	
Función	Eliminar trabajo registrado por el alumno.
Descripción	Permite al asistente eliminar los trabajos registrados por los alumnos.
Entradas	
Salida	Mensaje de confirmación.
Proceso	El asistente deberá buscar el trabajo del alumno y dar click en eliminar.
Restricciones	Solo podrá eliminar los trabajos que no han sido aprobados por el consejo.

Tabla 46. Especificación del requerimiento eliminar trabajos.

Actor: Asistente

Habilitar edición de trabajos	
Función	Habilitar la edición de los trabajos.
Descripción	Permite al asistente habilitar la edición de los trabajos registrados por los alumnos para que puedan realizar alguna modificación.
Entradas	Nombre de usuario al que se le activará la edición.
Salida	Mensaje de confirmación.
Proceso	El asistente deberá seleccionar la opción “habilitar edición”, e ingresar el usuario.
Restricciones	Solo se deberán activar la edición a trabajos que ya hayan sido aprobados por el consejo.

Tabla 47. Especificación del requerimiento habilitar edición de trabajos.

Actor: Asistente

Editar trabajos	
Función	Editar trabajo registrado por el alumno.
Descripción	Permite al asistente editar los trabajos registrados por los alumnos.
Entradas	Datos del trabajo: fecha de registro, título del trabajo, nombre del director de tesis, nombre de los asesores propietarios(tesis), nombre de los asesores suplentes(tesis), modalidad(monografía), comité de supervisión(monografía), resumen, observaciones.
Salida	Mensaje de confirmación.
Proceso	El asistente deberá seleccionar la opción “trabajos”, en seguida “editar” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrarán los cambios.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 48. Especificación del requerimiento editar trabajos.

Actor: Asistente

Envío de notificaciones	
Función	Envío de notificaciones
Descripción	Permite al asistente enviar notificaciones a los usuarios.
Entradas	Nombre del usuario y mensaje.
Salida	Mensaje de confirmación del envío.
Proceso	El administrador deberá seleccionar la opción enviar aviso, e ingresar el nombre de usuario y el mensaje.
Restricciones	Solo se podrá enviar notificaciones a los profesores y alumnos registrados en el sistema.

Tabla 49. Especificación del requerimiento envío de notificaciones.

Actor: Asistente

Generar oficios	
Función	Generar documentos de aprobación o edición de un trabajo.
Descripción	Permite generar oficios de aceptación o edición de un trabajo para el comité y el alumno.
Entradas	Tipo de oficio, Fecha actual, fecha de sesión del consejo (tesis), nombre de usuario del alumno (matrícula sin guión).
Salida	Confirmación de datos ingresados.
Proceso	El sistema desplegará el formulario para la generación de oficios en el cual se deberán de ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos requeridos se validarán los datos y se mostrara el oficio en pantalla con formato PDF.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 50. Especificación del requerimiento generar oficios (asistente).

Actor: Asistente

Generar Reportes	
Función	Mostrar estadísticas.
Descripción	Permite al asistente generar reporte estadísticos sobre los trabajos de titulación.
Entradas	Carrera y año.
Salida	Gráficas estadísticas con los resultados de la búsqueda.
Proceso	La asistente deberá seleccionar la opción de "reporte", posteriormente seleccionar una de las modalidades.
Restricciones	Se deberá seleccionar una de las opciones de búsqueda, ya sea por carrera o todos los registros forzosamente.

Tabla 51. Especificación del requerimiento generar reportes (asistente).

Actor: Asistente

Asignar Programa Educativo	
Función	Asignar un profesor a un Programa Educativo.
Descripción	Permite al asistente agregar a un profesor a un determinado Programa Educativo
Entradas	Nombre de usuario.
Salida	Mensaje de confirmación.
Proceso	El asistente deberá seleccionar la opción “Programa Educativo”, en seguida “agregar integrante” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el usuario.
Restricciones	Solo se deberán agregar a los profesores que ya tengan asignada una división y carrera.

Tabla 52. Especificación del requerimiento asignar Programa Educativo.

Actor: Asistente

Consultar registros	
Función	Consultar registros guardados en la base de datos.
Descripción	Permite al asistente realizar consultas de los registros guardados en la base de datos.
Entradas	Modalidad, consulta (campos de búsqueda).
Salida	Resultados de la búsqueda.
Proceso	El sistema desplegará el formulario en el cual los usuarios deberán seleccionar la modalidad de la búsqueda, posteriormente deberá seleccionar e ingresar los campos de búsqueda.
Restricciones	Se deberá seleccionar o ingresar como mínimo una opción de lo contrario el sistema mandara un mensaje de alerta.

Tabla 53. Especificación del requerimiento consultar registros (asistente).

Actor: Asistente

Agregar egresados	
Función	Agregar egresados de la división.
Descripción	Permite al asistente agregar egresados, ya sea por trabajo de titulación o por promedio.
Entradas	Nombre de usuario del alumno, nombre del presidente, nombre del secretario, nombre del vocal.
Salida	Mensaje de confirmación.
Proceso	El sistema desplegará el formulario en el cual se deberán ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el egresado.
Restricciones	Únicamente se podrán agregar egresados de nivel licenciatura.

Tabla 54. Especificación del requerimiento agregar egresados.

Actor: Profesor

Autenticarse en el sistema	
Función	Autenticarse en el sistema para poder hacer uso de él.
Descripción	Permite al profesor ingresar al sistema para hacer uso de su interfaz.
Entradas	Usuario, contraseña.
Salida	Interfaz de usuario.
Proceso	El profesor ingresa en el sistema su nombre de usuario y contraseña, el sistema validará si el usuario existe y si su contraseña es correcta. Si es así, el sistema mostrará al usuario la interfaz correspondiente a su perfil en el sistema.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 55. Especificación del requerimiento autenticarse en el sistema (profesor).

Actor: Profesor

Generar Reportes	
Función	Mostrar las estadísticas.
Descripción	Permite al profesor ver las estadísticas de los trabajos donde ha participado.
Entradas	año
Salida	Gráficas estadísticas con los resultados de la búsqueda.
Proceso	El profesor deberá seleccionar la opción de “reporte”, posteriormente ingresar el año que desea buscar.
Restricciones	

Tabla 56. Especificación del requerimiento generar reportes (profesor).

Actor: Profesor

Generar oficios	
Función	Generar documentos de la academia.
Descripción	Permite generar oficios de la academia.
Entradas	Nombre de usuario del alumno (matrícula sin guión).
Salida	Confirmación de datos ingresados.
Proceso	El sistema desplegará el formulario para la generación de oficios en el cual se deberán de ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos requeridos se validarán los datos y se mostrara el oficio en pantalla con formato PDF.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 57. Especificación del requerimiento generar oficios (profesor).

Actor: Profesor

Consultar registros	
Función	Consultar registros guardados en la base de datos.
Descripción	Permite al profesor, realizar consultas de los registros guardados en la base de datos.
Entradas	Modalidad, consulta (campos de búsqueda).
Salida	Resultados de la búsqueda.
Proceso	El sistema desplegará el formulario en el cual los usuarios deberán seleccionar la modalidad de la búsqueda, posteriormente deberá seleccionar e ingresar los campos de búsqueda.
Restricciones	Se deberá seleccionar o ingresar como mínimo una opción de lo contrario el sistema mandara un mensaje de alerta.

Tabla 58. Especificación del requerimiento consultar registros (profesor).

Actor: Administrador

Autenticarse en el sistema	
Función	Autenticarse en el sistema para poder hacer uso de él.
Descripción	Permite al administrador ingresar al sistema para hacer uso de su interfaz.
Entradas	Usuario, contraseña.
Salida	Interfaz de usuario.
Proceso	El administrador ingresan en el sistema su nombre de usuario y contraseña, el sistema validará si el usuario existe y si su contraseña es correcta. Si es así, el sistema mostrará al usuario la interfaz correspondiente a su perfil en el sistema.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 59. Especificación del requerimiento autenticarse en el sistema (administrador).

Actor: Administrador

Crear usuarios	
Función	Crear nuevos usuarios del sistema.
Descripción	Permite al administrador crear nuevos usuarios del sistema.
Entradas	Grado, nombres, apellidos, nombre de usuario, contraseña, perfil del usuario, tipo de cuenta.
Salida	Mensaje de confirmación.
Proceso	El administrador deberá seleccionar la opción "usuarios", en seguida "agregar" e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el usuario.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 60. Especificación del requerimiento crear usuarios (administrador).

Actor: Administrador

Editar usuarios	
Función	Editar usuarios del sistema.
Descripción	Permite al administrador editar a los usuarios del sistema.
Entradas	Grado, nombres, apellidos, nombre de usuario, contraseña, perfil del usuario, tipo de cuenta.
Salida	Mensaje de confirmación.
Proceso	El administrador deberá seleccionar la opción “editar” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el usuario.
Restricciones	Si los datos ingresados son incorrectos, el sistema mostrará un mensaje informando que los datos introducidos son incorrectos y solicitará que sean ingresados nuevamente.

Tabla 61. Especificación del requerimiento editar usuarios (administrador).

Actor: Administrador

Asignar división	
Función	Asignar división a los usuarios.
Descripción	Permite al administrador asignarles una división a los usuarios con perfil de asistente y profesor.
Entradas	Nombre de usuario y carrera (en caso de ser un profesor).
Salida	Mensaje de confirmación.
Proceso	El administrador deberá seleccionar la opción “usuarios”, en seguida “asignar división” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará el usuario.
Restricciones	

Tabla 62. Especificación del requerimiento asignar división (administrador).

Actor: Administrador

Nueva división	
Función	Crear una división.
Descripción	Permite al administrador crear una nueva división.
Entradas	Nombre de la división y abreviatura.
Salida	Mensaje de confirmación.
Proceso	El administrador deberá seleccionar la opción “nueva división” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará la nueva división.
Restricciones	

Tabla 63. Especificación del requerimiento nueva división.

Actor: Administrador

Nueva carrera	
Función	Crear una carrera.
Descripción	Permite al administrador crear una nueva carrera.
Entradas	Nombre de la división, nombre y abreviatura de la carrera.
Salida	Mensaje de confirmación.
Proceso	El administrador deberá seleccionar la opción “nueva carrera” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará la nueva carrera.
Restricciones	

Tabla 64. Especificación del requerimiento nueva carrera.

Actor: Administrador

Agregar director	
Función	Agregar a un director de división.
Descripción	Permite al administrador agregar un director de división.
Entradas	Grado, nombres, apellidos y división del director.
Salida	Mensaje de confirmación.
Proceso	El administrador deberá seleccionar la opción “directores”, en seguida “agregar” e ingresar los datos -descritos en la entrada-. Después de ser ingresados los datos se validarán y se registrará al director
Restricciones	Sólo se puede agregar a un director por división si se quiere añadir otro, se deberá eliminar el que se encuentra actualmente registrado.

Tabla 65. Especificación del requerimiento agregar director.

Actor: Administrador

Eliminar director	
Función	Eliminar a un director de división
Descripción	Permite al administrador eliminar a un director de división.
Entradas	
Salida	Mensaje de confirmación.
Proceso	El administrador deberá buscar al director y dar click en eliminar.
Restricciones	Si se elimina a un director, es necesario ingresar uno nuevo, ya que no se puede quedar sin director una división.

Tabla 66. Especificación del requerimiento eliminar director.

ANEXO B

Código fuente de la aplicación

El código fuente de la aplicación se encuentra en el siguiente [vínculo](#).

ANEXO C

Manuales de usuario

Los manuales de usuario se encuentran en el siguiente [vínculo](#).

ANEXO D

Cuestionario usabilidad (asistentes)

Cuestionario de Usabilidad (Asistentes)

Responde las siguientes preguntas de acuerdo a lo que hayas experimentado durante las pruebas realizadas al sistema.

1.- ¿Cómo consideras el nivel de usabilidad en este sistema? Califica de 0 a 10:

MUY BUENO

2.- ¿Te fue difícil realizar las tareas asignadas dentro del sistema?

UN POCO

¿Por qué?

POC DESCONOCIMIENTO DEL SISTEMA

3.- ¿Crees que este sistema puede ser implementado en la vida cotidiana?

SI

4.- La interacción que tuviste con la interfaz del sistema. ¿Fue agradable?

SI

5.- ¿Qué te gustaría que tuviera la interfaz para hacerla más agradable?

NO SÉ

6.- ¿Qué otras tareas se le pueden agregar al sistema?

NO SÉ, NO SE ME OCURRE NADA POR AHORA

7.- ¿Cómo consideras el nivel de funcionalidad del sistema? (Actividades que puedes realizar con el sistema) Califica del 0 al 10: 10

8.- ¿Cómo consideras que fue la velocidad del software? (Buena/Regular/Mala)

9.- ¿Qué características positivas pudiste ver dentro del sistema?

AHORRO DE TIEMPO

10.- ¿Qué características negativas pudiste ver dentro del sistema?

NO LE VI

11.- Escribe un comentario de manera general sobre tu experiencia en el uso de este sistema.

EN EL OFICIO DE TESTS Q' SE GENERA, CREO ESTARIA BIEN ELIMINAR EN LA PARTE DONDE SE INDICAN LAS COPIAS, LAS

Cuestionario de Usabilidad (Asistentes)

Responde las siguientes preguntas de acuerdo a lo que hayas experimentado durante las pruebas realizadas al sistema.

1.- ¿Cómo consideras el nivel de usabilidad en este sistema? Califica de 0 a 10:

9

2.- ¿Te fue difícil realizar las tareas asignadas dentro del sistema?

No

¿Por qué?

3.- ¿Crees que este sistema puede ser implementado en la vida cotidiana?

Si me gustaria

4.- La interacción que tuviste con la interfaz del sistema. ¿Fue agradable?

Si

5.- ¿Qué te gustaría que tuviera la interfaz para hacerla más agradable?

Está buena

6.- ¿Qué otras tareas se le pueden agregar al sistema?

Está bien así

7.- ¿Cómo consideras el nivel de funcionalidad del sistema? (Actividades que puedes realizar con el sistema) Califica del 0 al 10: 10

8.- ¿Cómo consideras que fue la velocidad del software? (Buena/Regular/Mala)

Buena

9.- ¿Qué características positivas pudiste ver dentro del sistema?

- Acortar el tiempo
- Guía para obtener los elementos con sus funciones
- Mayor control

10.- ¿Qué características negativas pudiste ver dentro del sistema?

11.- Escribe un comentario de manera general sobre tu experiencia en el uso de este sistema.

Está muy bien para llevar un mejor control y un historial de los registros.