

UNIVERSIDAD POLITÉCNICA DEL ESTADO DE MORELOS



Upemor
Universidad **Politécnica**

“Tutorial Web con Sistema de Tres Capas, Para la enseñanza de la Programación Distribuida con Paso de Mensajes de Estructuras Complejas”

Tesina

Que para obtener el título de:

INGENIERO EN INFORMÁTICA

Presenta:

JOSÉ MANUEL PATIÑO GUTIÉRREZ

Directores de tesina:

M. en C Roberto Enrique López Díaz

Dr. Marco Antonio Cruz Chávez

Jiutepec, Morelos

SEPTIEMBRE - MARZO 2015

ÍNDICE

Capítulo 1. Introducción	1
1.1 Antecedentes.....	1
1.2 Planteamiento del problema.....	2
1.3 Panorama general del proyecto.....	2
1.4 Objetivos.....	3
1.5 Justificación.....	3
1.6 Alcances y limitaciones.....	4
1.7 Metodología.....	5
1.7.1 SCRUM.....	5
1.7.2 Descripción de la metodología.....	5
1.7.3 Justificación de la metodología.....	6
1.8 Organización de la Tesina.....	7
Capítulo 2. Análisis	9
2.1. Introducción.....	9
2.2. Restricciones de diseño.....	9
2.3. Requisitos funcionales.....	10
2.3.1. Requisitos nominales.....	10
2.3.2. Requisitos no nominales.....	18
2.4. Requisitos de interfaz.....	20
2.5. Requisitos de calidad.....	20
2.6. Requisitos de evolución.....	21
2.7. Requisitos de proyecto.....	21
2.8. Requisitos de soporte.....	21

Capítulo 3 Diseño	23
3.1 Arquitectura del sistema.....	23
3.1.1 Arquitectura lógica del sistema	23
3.1.2 Patrón de Modelo-Vista-Controlador (MVC).....	24
3.2 Diseño de requisitos funcionales	25
3.3 Diseño de la Interfaz de usuario	38
3.4 Diseño de la Base de datos	41
3.4.1 Diagrama Entidad-Relación	41
3.4.2 Diccionario de datos	42
Capítulo 4 Implementación del sistema	46
4.1 Arquitectura física	46
4.1.1 Capa de presentación.....	46
4.1.2 Capa de negocio	47
4.1.3 Capa de datos.....	47
4.2 Tecnologías empleadas	48
4.3 Solución de problemas de implementación.....	51
4.3.1 Botón de menú	51
4.3.2 Excepción "java.sql.SQLException: No suitable driver found for jdbc"	53
4.3.3 Insertar tutoriales	56
4.4 Documentación de código fuente	57
4.4.1 Inicio de sesión múltiple.....	57
4.4.2 Perfil de usuario	59
4.4.3 Panel de control.....	60
4.4.4 Añadir tutoriales	64
Capítulo 5 Pruebas e implantación	69
5.1 Ambiente de pruebas	69

5.2 Base de datos de pruebas	71
5.3 Ejecución del plan de pruebas	75
5.4 Bitácora de implantación	93
Capítulo 6 Conclusiones y trabajo futuro	95
6.1 Conclusiones	95
6.2 Trabajo futuro	96
Bibliografía	98

Capítulo 1. Introducción

En este capítulo se describen características, objetivos del sistema, además de algunas cuestiones generales, de donde surge, por qué, y en qué precisa ayudar, además de definir el proceso de desarrollo mediante la elección de una metodología adecuada, al final del capítulo se hablará de groso modo de la organización de la tesina titulada: "Tutorial Web con Sistema de Tres Capas, Para la enseñanza de la Programación Distribuida con Paso de Mensajes de Estructuras Complejas".

1.1 Antecedentes

El Centro de Investigación en Ingeniería y Ciencias Aplicadas (CIICAp), dependiente de la Universidad Autónoma del Estado de Morelos (UAEM). Genera de manera constante, proyectos de Investigación en Ciencia y Tecnología, en busca de la trascendencia en el área de Ingeniería y Ciencias Aplicadas realizando vinculación, actualización e innovación orientada, para satisfacer eficazmente las expectativas de los sectores educativos, productivo y social, mediante el desarrollo de tecnología, aplicación y generación del conocimiento mismo que sustenta la misión del CIICAp. De acuerdo con esto, el cuerpo académico que hace la optimización y software coordinado por el Dr. Marco Antonio Cruz Chávez, requiere de un recurso humano capacitado para la elaboración de proyectos para cumplir con la visión de este centro de investigación. El cuerpo académico "InterDES Optimización y Software" adscrito al Centro de Investigaciones en Ingeniería y Ciencias Aplicadas de la UAEM en al área de CIENCIAS EXACTAS E INGENIERÍA tiene como función principal: la creación, mejora y aplicación de métodos y algoritmos en optimización combinatoria, diseño y análisis de métodos y algoritmos, y aplicación de métodos y algoritmos mejorados, área y lugar en la que se desarrollará el sistema.

1.2 Planteamiento del problema.

El CIICAp en el área de ciencias exactas e ingeniería cuenta con un cuerpo académico dedicado a la investigación y desarrollo de algoritmos de programación distribuida con paso de mensajes (MPI). El cuerpo académico además de los alumnos de la institución al realizar sus trabajos, investigaciones, proyectos, etc. se enfrentan con las dificultades al utilizar el clúster o la grid según lo amerite ya que en algunos casos no cuentan con un claro conocimiento de cómo opera la grid, o existe desconocimiento de su funcionamiento, o pueda existir la necesidad de mantenimiento o soporte y que no se encuentre funcionando correctamente.

A nivel de programación por la complejidad misma de la grid, suele haber problemas a la hora de ejecutar código, además de que existen varios compiladores que se usan de diferente manera, por lo que se necesita consultar información, cabe resaltar que muchos de los problemas y los principales son por algunas funciones del lenguaje c usando las librerías de MPI o similares, en varias ocasiones el código está preparado para ser ejecutado en compiladores de manera secuencial por lo que tienen que convertir el código para que se ejecute en el clúster es decir con paralelismo, pero la dificultad al ser una herramienta muy nueva está en no tener la información íntegra lo cual gasta tiempo en búsquedas y consultas en otras fuentes diversas o varias, por lo que una manera de documentar y mantener íntegro todo en un mismo lugar es presentar la información en tutoriales, y el hacerlo de manera web permitiría la portabilidad del mismo.

1.3 Panorama general del proyecto

En el año 2010 el Dr. Marco Antonio Cruz Chávez a través de Cuerpo Académico en Optimización y Software propuso el proyecto de realizar una grid, la "Grid Morelense de Alto Rendimiento" y en conjunto con otras universidades del estado de Morelos (Universidad Autónoma del Estado de Morelos y la Universidad Politécnica del Estado de Morelos) así como del Instituto Tecnológico de Veracruz ubicada en el estado de Veracruz que ahora está actualmente funcionando y cada institución antes mencionada cuenta con un clúster que conforman la grid.

Actualmente se realizan pruebas de complejidad, eficacia y eficiencia de nuevas propuestas para hiper-heurísticas en búsquedas de vecindades para mejorar el desempeño de estas heurísticas. Mismas en donde investigadores, profesores y alumnos, hacen uso de la grid, a través de cada nodo clúster que la conforman, es decir la UPEMOR (Universidad Politécnica del Estado de Morelos), hace uso de la grid por medio del clúster que hay en la UPEMOR, por otra parte, en la UAEM (Universidad Autónoma del Estado de Morelos) utilizan la grid por medio del clúster ubicado en el la UAEM, así mismo para la Instituto Tecnológico de Veracruz.

1.4 Objetivos

Los objetivos ayudan a ver lo que se que se quiere lograr, tienen que ser enfocados a una meta, además de que deben ser claros, en este caso el objetivo es el siguiente:

Realizar un sistema web de tres capas que ayude a la enseñanza y aprendizaje de MPI en lenguaje C y que permita realizar evaluaciones.

1.5 Justificación

El sistema web que se presenta en la tesina tiene como justificación servir para el fácil aprendizaje y análisis de código MPI dirigido a estudiantes de posgrado en ingeniería que requieran en sus proyectos de investigación, la aplicación y desarrollo de cómputo de alto desempeño, ya que el sistema permitirá a los estudiantes mejoren sus conocimiento y aprendizaje a través de este sistema. Es decir, será una herramienta donde el usuario puede conocer, aprender y evaluar sus conocimientos. Al cubrir la necesidad se busca también ahorrar tiempo en búsquedas sin certeza de resultados, y utilizar ese tiempo en aprender de una mejor manera, de poder evaluar sus conocimientos.

1.6 Alcances y limitaciones

A continuación se muestran los alcances y las limitaciones del presente trabajo, mismas que ayudarán a establecer los objetivos.

Alcances:

- El sistema contará con los módulos de subir, actualizar, editar, eliminar, contenido, además de evaluaciones.
- Las evaluaciones serán gestionadas por el administrador, es decir, el administrador las generará, mientras que el usuario alumno solo podrá evaluar sus conocimientos.
- Se desarrollará una base de datos adecuada para el sistema, que debe ser en un servidor de BD (base de datos).
- La interfaz desarrollada será en tecnologías adecuadas y nuevas como HTML5 y JS con CSS3, lo que permite una mejor experiencia al usuario, por no necesitar plugins, etc.
- Se implementará un CMS (Content Management System), que servirá para que el usuario administrador pueda gestionar el contenido de los tutoriales.

Limitaciones:

- No se realizarán manuales (usuario, técnico etc.) para el sistema.
- El sistema solo contará con una arquitectura de tres capas.
- El tiempo de desarrollo consta de solo 4 Meses en un periodo de Septiembre a Diciembre del 2014.

1.7 Metodología

La metodología es muy importante ya que permite tener un mejor desarrollo, es decir, tener una metodología adecuada ayudará a obtener mejores resultados al final. La metodología que se ocupara para este sistema será la SCRUM.

1.7.1 SCRUM

SCRUM es una metodología de desarrollo ágil para desarrollo de software, en sus inicios fue basado en un conjunto de buenas practicas, para obtener la mejor calidad posible en los resultados de los proyectos, además de, mantener comunicación con todo el equipo de desarrollo y el cliente[1].

1.7.2 Descripción de la metodología

El uso de esta metodología implica ciertas actividades, y documentación que posteriormente se agregará como anexo A. Las actividades son las siguientes a partir de nuestro producto:

1. Se necesita saber los requisitos del producto, tomadas de las perspectivas del usuario final, conocidas como historias de usuario.
2. La colección de estas historias es llamado BACKLOG o lista de deseos.
3. Después se definen los roles del equipo, que esta conformado por:
 - 3.1. Product owner: se encarga de decidir que historias de usuario van a usarse.
 - 3.2. Scrum master: el cual es encargado de mantener al equipo unido además de verificar que el equipo cuente con las herramientas necesarias para el desarrollo del producto, así como realiza la planeación de entrega del producto y organizar las reuniones diarias.
 - 3.3. Developer: se trata de un equipo (la cantidad depende de lo robusto del sistema) que está encargado del desarrollo del producto.
 - 3.4. Tester: encargado (al igual que el developer la cantidad depende de lo robusto del sistema) de realizar las pruebas del producto y verificar que todo funcione bien en el sistema.
 - 3.5. Ejecutivos: se dedican a cotizar y realizar los procesos administrativos.
4. Release planing: consta de las historias de usuario que se van a liberar, y que deben pasar ahora al BACKLOG de liberación. El equipo prioriza las historias y estima tiempos para cada una de ellas. Existen historias muy grandes por lo

que se dividen en pequeñas historias para hacerlas más manejables y la suma de todas estas historias de usuario del BACKLOG de liberación forman el tiempo estimado que tardará en haber una liberación.

5. Sprints: son el hito de las historias de usuario del BACKLOG de liberación, que están organizados en periodos de 2 días, hasta 30 días dependiendo del ciclo de liberación del producto. Entre más corto es el ciclo de liberación más corto debe ser el sprint. El conjunto de sprints de liberación deben ser terminadas dentro del tiempo estimado de lo contrario es un indicador que algo no cumple con lo estimado y puede afectar el proyecto por lo que deben verificarse.
6. Burndown chart: es una herramienta de visibilidad de proyectos que provee la cantidad de trabajo realizado día a día. Esta herramienta es opcional, debido a que solo ayuda con la perspectiva del proyecto y no altera el resultado al no usarla.
7. Reunion diaria: se listan las tareas terminadas de la reunión anterior (sí la hay), además de los obstáculos inconvenientes y la manera de solucionar los mismos. Al finalizar se da una retrospectiva de las reuniones del sprint para ver que estuvo bien y que estuvo mal.

En la figura [1], se muestra el flujo de las iteraciones y componentes antes mencionados.

1.7.3 Justificación de la metodología.

SCRUM puede ser aplicado para desarrollo de sistemas web, ya que entre sus ventajas se encuentran: que permite hacer entregas mensuales o quincenales; que el proceso es iterativo, y con cada iteración se puede determinar el estado del producto final; y la flexibilidad que tiene permite al usuario regular sus prioridades. De manera que al tener como limitación el tiempo de desarrollo, es conveniente una metodología que permita entregas rápidas. SCRUM por sus características es una opción adecuada, solo que no se utilizará el burndown chart.

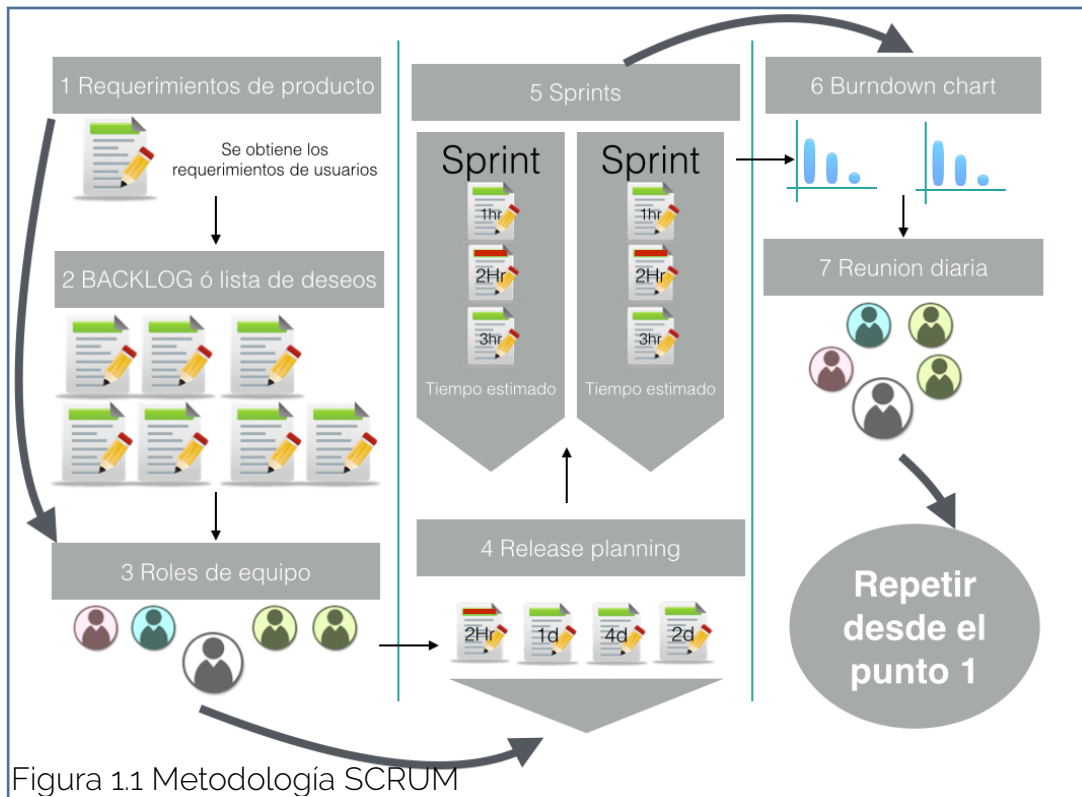


Figura 1.1 Metodología SCRUM

1.8 Organización de la Tesina

Este trabajo constará de los siguientes capítulos:

Capítulo 1. Introducción:

Se describen las características del sistema, además como surge y los objetivos a donde se quiere llegar.

Capítulo 2. Análisis:

En este capítulo se muestran los requisitos, así como la obtención de los mismos, y se realizan algunos diagramas que forman parte del análisis, para el desarrollo posterior del sistema.

Capítulo 3. Diseño:

En el capítulo de diseño se muestra el arte de la creación de sistemas de software, es decir el planteamiento de la arquitectura, explicación del diseño lógico y físico, y el diseño de la base de datos.

Capítulo 4. Desarrollo:

Muestra el proceso de desarrollo la justificación de las tecnologías, y algunos problemas afrontados.

Capítulo 5. Pruebas e Implantación:

En este capítulo se desarrollan pruebas del sistema, con el fin de otorgar una mejor calidad al sistema y fijar las bases de la implantación.

Capítulo 6. Conclusiones:

Describen los logros y resultados obtenidos, además de lo que queda por hacer, y qué se puede mejorar en el sistema.

Capítulo 2. Análisis

2.1. Introducción

Para empezar a desarrollar, es necesario primero identificar características más específicas del sistema para ello se tiene que indagar con el cliente. El primer lugar donde se tiene que buscar los requisitos es con el cliente solicitante, y luego identificar a los usuarios finales (si es necesario) para obtener más información, en el CIICAp el Dr. Marco Antonio Cruz Chavez es a quien primero se recurrirá para obtener la información, debido al poco tiempo que dispone se necesita que la técnica para la obtención de requisitos sea muy ágil, y de manera rápida.

Existen muchas técnicas para saber las necesidades del usuario, algunas proporcionan de una manera gráfica la mejor obtención de los requisitos, sin embargo la comunicación con el cliente y el usuario final ayudan a detallar de una mejor manera estas necesidades[2]. Entender las necesidades del cliente es fundamental antes de que se comience con un diseño o desarrollo, lamentablemente el proceso es muy difícil debido a que los requisitos tienden a ser muy cambiantes o inestables, sin embargo, la metodología de desarrollo utilizada lo acepta ya que se filtrarán los requisitos.

La técnica que se utilizará para la obtención de requisitos es "la más tradicional de las técnicas de obtención y consiste en reuniones analista-interesado en las cuales se suceden preguntas y respuestas para extraer el dominio de la aplicación" [3].

2.2. Restricciones de diseño

Las restricciones de diseño ayudan a identificar cuales van a hacer las especificaciones de desarrollo, es decir identificar las necesidades que se tienen para la orientación a desarrollar con algún lenguaje en específico.

RD 1. Sistema web

Descripción: El sistema de tutoriales queda limitado a un sistema web. Es decir que el sistema debe correr bajo un servidor web.

RD 2. Tecnologías de desarrollo

Descripción: Las tecnologías de desarrollo quedan a disposición del desarrollador, la restricción está en que ésta debe ser orientado al uso de tecnologías nuevas, de vanguardia y seguras.

RD 3. Datos e información.

Descripción: El sistema gestor de base de datos (SGBD) deberá ser mysql server, debido a que estará en un servidor, y no se necesita una base de datos de licencia comercial o no gratuita.

Importancia: Esencial.

RD 4. Diseño de la interfaz

Descripción: La tecnología usada para el diseño queda a consideración del desarrollador pero ésta se enfocará en tecnologías de vanguardia y nuevas.

Importancia: Esencial.

2.3. Requisitos funcionales

En esta sección se muestran los requisitos nominales y no nominales del sistema, ayudarán a comprender las necesidades del usuario en cuanto a la funcionalidad y comportamiento del sistema, así como en situaciones especiales, como errores o advertencias.

2.3.1. Requisitos nominales

Se muestran los requisitos del sistema nominales para comportamiento y funcionalidad.

Panel de control

FN. 1 Panel de control

Descripción El sistema permitirá al administrador gestionar los tutoriales y las evaluaciones, de formas separadas, es decir existirá un módulo para gestionar tutoriales y otro módulo para gestionar exámenes, además existirá un usuario súper administrador que podrá gestionar a los usuarios alumnos.

Importancia Esencial.

Validez:

Medible: Es necesario para el sistema contar con el panel de control ya que permitida añadir contenido y usuarios.

Alcanzable: Para poder acceder al panel de control, solo necesita iniciar sesión y el sistema lo validará, obteniendo así que sí éste es un administrador mostrará la opción de acceder al panel de control.

Relevante: Este requisito es muy importante ya que permite generar contenido nuevo.

FN. 2 Gestión de tutoriales

Descripción Este requisito permitirá habilitar la creación y modificación de contenido nuevo en cuanto a tutoriales se requiere, estos deberán ser creados a través de un formulario que permitirá hacer de manera más sencilla su creación, al igual para poder modificar o eliminarlo.

Importancia Esencial.

Validez:

Medible: El sistema necesita el apartado de gestión de tutoriales ya que permite lograr el objetivo y la función principal del sistema.

FN. 2 Gestión de tutoriales

- Alcanzable: La gestión de tutoriales será un privilegio del usuario administrador, y para acceder a este apartado se hace mediante el panel de control.
- Relevante: Este requisito es medular en la creación de este sistema, y debido a su importancia este no puede faltar.

FN. 3 Gestión de evaluaciones

Descripción La gestión de exámenes permite al administrador crear, modificar o eliminar exámenes, para poder crear exámenes se necesita de un formulario que se debe llenar pero que permite crear exámenes de una manera más fácil. Así mismo mediante un formulario permitirá modificar o eliminar los exámenes.

Importancia Esencial.

Validez:

- Medible: El sistema requiere de realizar evaluaciones a los alumnos para poder ver como se comportan en cuanto al aprendizaje de los alumnos.
- Alcanzable: Para poder acceder al apartado de gestión de exámenes o evaluaciones, se requiere haber iniciado sesión como administrador posteriormente ir al panel de control.
- Relevante: Este requisito es esencial pero es un privilegio de los usuarios de tipo alumnos del sistema, además permitirá al administrador darse cuenta de los avances.

FN. 4 Gestión de usuarios

Descripción	La gestión de usuarios permite al administrador dar de alta o baja a usuarios de tipo administradores o alumnos, pero esta función solo aplica para un usuario súper usuario (los alumnos si pueden darse de alta como usuarios de tipo alumno). Los tipos de usuarios son Administrador, Alumno, e invitado.
Importancia	Esencial.
Validez:	
Medible:	En el sistema es necesario tener jerarquías de usuarios debido a que los tutoriales solo los pueden realizar personal autorizado, y no los alumnos, lo que permite una mayor calidad de contenido.
Alcanzable:	Para poder acceder a este apartado el usuario súper administrador debe iniciar sesión después, dirigirse al apartado de panel de control, para ello el sistema ya lo reconocerá como súper administrador, pero acceder al módulo de gestión de usuarios pide ingresar nuevamente la contraseña del usuario.
Relevante:	Este requisito es esencial ya que afecta el contenido y objetivo del sistema, hace seguro el uso de este, además de reflejar el flujo de usuarios que se tengan en el sistema (solo usuarios alumno y administradores).

FN. 5 Historial de alumnos.

Descripción En el sistema el súper usuario puede también revisar el avance de los alumnos es decir consultar de acuerdo a un alumno su historial de exámenes.

Importancia Esencial.

Validez:

Medible: El sistema debe permite ver el avance solo para el súper usuario, ya que le puede servir de ayuda para revisar su experiencia, solo se almacena las calificaciones del examen, el número de veces que lo ha realizado y los exámenes que este ha hecho. Cabe resaltar que el sistema evalúa los exámenes con calificación mínima de 8.

Alcanzable: Para visualizar el historial de alumnos el súper usuario deberá de iniciar sesión y posteriormente dirigirse al panel de control en el apartado de historial de alumnos.

Relevante: Este requisito es esencial debido a que abre el panorama al administrador general del sistema a ver el avance de los alumnos.

Funciones Generales

FN. 6 Registro de usuarios

Descripción El sistema contará con la capacidad para que usuarios de tipo invitado puedan registrarse como alumnos, así poder realizar evaluaciones de sus conocimientos.

Importancia Esencial.

Validez:

Medible: Los usuarios dependiendo su tipo necesitan ingresar con un usuario y contraseña, que serán validados para poder entrar a un perfil.

Alcanzable: El sistema contará con un apartado para el registro de usuarios que ayudará a mantener las funciones que cada uno de ellos tiene permitido.

Relevante: Es necesario contar con el registro de usuarios ya que permite al sistema guardar el historial de las evaluaciones y al alumno evaluar sus conocimientos.

FN. 7 Perfil de usuario de tipo alumno.

Descripción El sistema una vez tenga un usuario de tipo alumno podrá acceder a un perfil, en donde muestre sus calificaciones de los exámenes.

Importancia Esencial.

Validez:

Medible: El sistema podrá contar con un perfil que mantenga su historial de exámenes.

Alcanzable: El sistema tendrá un apartado para visualizar el perfil, cabe mencionar que si no se esta registrado como alumno no existirá tal apartado.

Relevante: La presencia del perfil ayudará al alumno a tener su historial de exámenes.

FN. 8 Mostrar tutoriales

Descripción	El sistema cuenta con un módulo para visualizar los tutoriales que existan, cada uno contará con su título, su contenido y estarán basados con la ayuda de la herramienta turbodemo.
Importancia	Esencial.
Validez:	
Medible:	Todos los usuarios tienen acceso a visualizar tutoriales, el sistema permitirá navegar a dichos usuarios por el apartado de los tutoriales.
Alcanzable:	El sistema mostrará un tutorial en relación al tema, se mostrará uno por página web, para llegar a el tutorial debió haberlo elegido del índice o de una búsqueda.
Relevante:	Es el objetivo principal del sistema, es por eso que debe de existir en él.

FN. 9 Índice de tutoriales

Descripción	El sistema tendrá un índice de temas el cual deberá de organizarse de menor a mayor dificultad.
Importancia	Condicional.
Validez:	
Medible:	El sistema tendrá un índice dependiendo a los tutoriales que haya. Estos deben corresponder a los tutoriales existentes.
Alcanzable:	Existirá un módulo específicamente para visualizar el índice.
Relevante:	Es necesario tener un índice ya que muestra los temas y su complejidad, debido a que están ordenados de menor a mayor dificultad.

FN. 10 Buscar tutoriales o exámenes.

Descripción	El sistema deberá contener una barra de búsqueda que permita encontrar tutoriales, y exámenes.
Importancia	Opcional.
Validez:	<p>Medible: El usuario podrá ingresar al apartado de índices o al de evaluaciones (exámenes), en donde observará la barra de búsqueda.</p> <p>Alcanzable: El sistema contará con la barra de búsqueda ubicado en el área de índice de tutoriales, y en evaluaciones, en ambos casos buscará según un filtro: exámenes, tutoriales o ambas.</p> <p>Relevante: Se trata de un requisito no fundamental debido a que la existencia o no de esta, no altera los tutoriales o las evaluaciones.</p>

FN. 11 Aplicar Evaluaciones

Descripción	El sistema deberá contener un módulo para realizar exámenes de acuerdo a el tema que el usuario elija. Deberá tener un tiempo determinado para realizarlo, según la complejidad del tutorial al que corresponda.
Importancia	Esencial.
Validez:	<p>Medible: El usuario de tipo alumno y superiores, pueden realizar exámenes que permitan medir su conocimiento acerca de un tema. la calificación mínima aprobatoria será de 8 tomando en cuenta que las evaluaciones se miden de 0 a 10 y solo serán 10 preguntas por examen de tutorial.</p> <p>Alcanzable: El usuario podrá acceder a realizar exámenes a través del apartado de evaluaciones en el sistema, después de haber iniciado sesión.</p>

FN. 11 Aplicar Evaluaciones

Relevante: Este requisito es necesario para cumplir con las funciones del sistema y sus objetivos por lo que no puede faltar.

2.3.2. Requisitos no nominales

Se muestran los requisitos del sistema no nominales para situaciones especiales o de error.

FF. 1 Validaciones

Descripción Para todos los formularios del sistema, se deben validar los datos introducidos, mediante filtros adecuados a las características de esos datos.

Importancia Esencial.

Validez:

Medible: Validar los formularios es una ayuda para identificar que el sistema tenga un poco más de calidad en el contenido.

Alcanzable: Al acceder a un tutorial es necesario que al momento de llenarlo, el sistema valide la consistencia de los datos.

Relevante: El requisito permitiría alcanzar un mayor valor en la calidad.

FF. 2 Advertencias

Descripción Advertir al usuario de manera adecuada cuando un usuario esta accediendo o realizando alguna acción, permite un mejor uso, además, acompañar las advertencias con un consejo puede ayudar al usuario a mejorar el uso del sistema.

Importancia Esencial.

Validez:

FF. 2 Advertencias

- Medible: El sistema necesita ayudar al usuario a aprender mejor, por lo que las advertencias permiten orientarlo en el sistema.
- Alcanzable: Cuando un usuario realice una acción como gestionar, o buscar puede asesorar al usuario con una advertencia que permite realizarlo de una mejor manera.
- Relevante: Este requisito permite mejorar la calidad por lo que es esencial mostrar advertencias adecuadas.

FF. 3 Notificaciones de exámenes

Descripción Cuando un usuario de tipo alumno realice una evaluación se necesita mostrar notificaciones de tiempo, de inicio, de fin, y las instrucciones.

Importancia Esencial.

Validez:

- Medible: El alumno cuenta con un tiempo determinado para realizar evaluaciones por lo que es necesario que éste identifique con cuánto tiempo cuenta y en que momento inicia y termina.
- Alcanzable: El sistema deberá mostrar en el módulo de evaluaciones una notificación que oriente al usuario a saber lo que sucede y pasa en el transcurso de su evaluación.
- Relevante: Este requisito permite mejorar la calidad del sistema, por lo que es un requisito importante cuando se realicen las evaluaciones.

2.4. Requisitos de interfaz

Los requisitos de interfaz ayudan a comprender la interacción del sistema con otros sistemas o con los usuarios.

IN.1 Interfaz independiente del CIICAp

Descripción: La interfaz puede ser independiente de la página que actualmente existe del CIICAp, la restricción está basada en que debe ser con colores que muestren seriedad y ayuden a no distraer a los usuarios de sus objetivos.

IN. 2 Estilos de interfaz

Descripción: El estilo de interfaz se utilizará mediante hojas de estilo CSS3 debido a que es una tecnología de vanguardia y más flexible que otras popularmente usadas como la tecnología Flash.

IN. 3 Interfaz del panel de control

Descripción: El sistema que contará con un panel de control, debe ser adecuado para que se intuya qué se está haciendo, es decir la interfaz debe estar adecuada para ayudar al usuario a cumplir sus objetivos.

2.5. Requisitos de calidad

Los requisitos de calidad ayudan al usuario a tener una mejor experiencia con el sistema.

CA.1 Usabilidad

Descripción: La usabilidad del sistema es un termino informático que ayuda a la interacción usuario-sistema y la restricción que el sistema debe cumplir es que ésta debe ser ágil y sencilla.

CA. 2 Calidad

Descripción: La calidad del sistema esta comprendida por los aspectos de la interfaz, por el desarrollo de los tutoriales y exámenes además de las validaciones, notificaciones y advertencias correspondientes.

CA. 3 Experiencia de usuario.

Descripción: En el desarrollo el sistema debe aplicar User Experience, lo cual es una forma de diseño web orientado al análisis de idea del negocio o producto, interfaz y contenido.

2.6. Requisitos de evolución

En esta sección se identifica los requisitos de evolución los cuales sirven e identifican la capacidad de adaptación del sistema, como la capacidad de poder hacerse más robusto sin perder la calidad.

Titulo: Ev.1 Escalabilidad

Descripción: El sistema tiene la opción de ser escalable para trabajos futuros sin embargo el uso de tecnologías de vanguardia ayudan a tener un mayor grado de escalabilidad a futuro, permitiendo añadir como parte de un sistema mas grande, o simplemente para hacer más robusto este mismo.

2.7. Requisitos de proyecto

Estos requisitos ayudan a saber las necesidades del usuarios relacionadas al proceso de desarrollo del sistema.

Titulo: PR.1 Tiempo de entrega del proyecto.

Descripción: El tiempo de entrega del proyecto será aceptable entre el periodo Diciembre 2014 - Febrero 2015

2.8. Requisitos de soporte

En esta sección se muestran los requisitos de soporte que servirán para la identificación de los requisitos para uso o instalación del sistema de tutoriales.

Titulo: SO. 1 Servidor HTTP

Descripción: Para la implantación del sistema se necesita un servidor http, el cual está de manera física, manejando en el peticiones http.

Titulo: SO. 2 Servidor MySql

Descripción: Para la implantación de la base de datos del sistema se tendrá un servidor físico y solo con el servicio del gestor de base de datos.

SO. 3 Equipo de desarrollo

Descripción: Para llevar a cabo el desarrollo del sistema, se cuenta con una computadora que permitirá la codificación y pruebas del sistema de manera local.

Capítulo 3 Diseño

Este capítulo se enfoca en el diseño y muestra la arquitectura del sistema, la transformación de los requisitos obtenidos a un diseño del sistema, comprendiendo las funciones, interfaces y diagramas de la base de datos.

3.1 Arquitectura del sistema

La arquitectura del sistema comprende la organización fundamental de los componentes del sistema desde dos aspectos, el lógico y el físico, comprende conceptos como su ambiente, la relación con otros sistemas, y ayuda a futuro con la evolución e inclusive mantenimiento del sistema. En el siguiente punto 3.1.1 se detallan la arquitectura lógica del sistema. La arquitectura de software se puede interpretar partiendo de los principios de la arquitectura como tal, ya que en ambos tipos de arquitectura se trata de hacer los planos para la construcción o desarrollo de algo, por ejemplo una casa o un edificio; en el caso de este trabajo, la arquitectura de software ayudará en el desarrollo del sistema, por medio de patrones de diseño.

Los patrones de diseño se relacionan con el estilo, en el desarrollo de la solución de los sistemas de información, en este caso los patrones funcionan como base de dividir o estructurar el sistema según lo requiera o se necesite, es decir crean sub-sistemas con responsabilidades además de también crear reglas de negocio (aunque no necesariamente)[4].

3.1.1 Arquitectura lógica del sistema

Los dos aspectos que existen, el aspecto lógico o físico, pueden fácilmente interpretarse como una analogía de software y hardware respectivamente, ya que el aspecto de la arquitectura lógica del sistema consta de estructurar la parte dedicada al software, en el sentido que debe ocupar algún patrón arquitectónico dependiente de las necesidades, un ejemplo de ello puede ser que la interfaz queda independiente de procesar la información y de almacenar la información. Mientras que el aspecto físico estructura la parte física, es decir algún patrón

arquitectónico para ello pero de manera física, un ejemplo de ello sería mantener separados un servidor web de un servidor dedicado a la base de datos.

3.1.2 Patrón de Modelo-Vista-Controlador (MVC)

El modelo-vista-controlador o MVC (Model-View-Controller) es un patrón de diseño arquitectónico que tomaremos para este sistema, debido a su división lógica que prevé el modelo, se presta para trabajar con la arquitectura física del sistema (En niveles o capas, de la cual se trata en el capítulo 4.1).

El modelo Vista Controlar consiste en 3 tipos de objetos, los cuales son: El Modelo es el objeto de aplicación que se encarga de almacenar y tratar con los datos que el sistema genere, la Vista que es la pantalla de la interfaz, y el Controlador que define la forma en que la interfaz de usuario reacciona con las entradas de datos [5]. En la figura 3.1, se puede observar gráficamente como esta organizado dicho patrón.

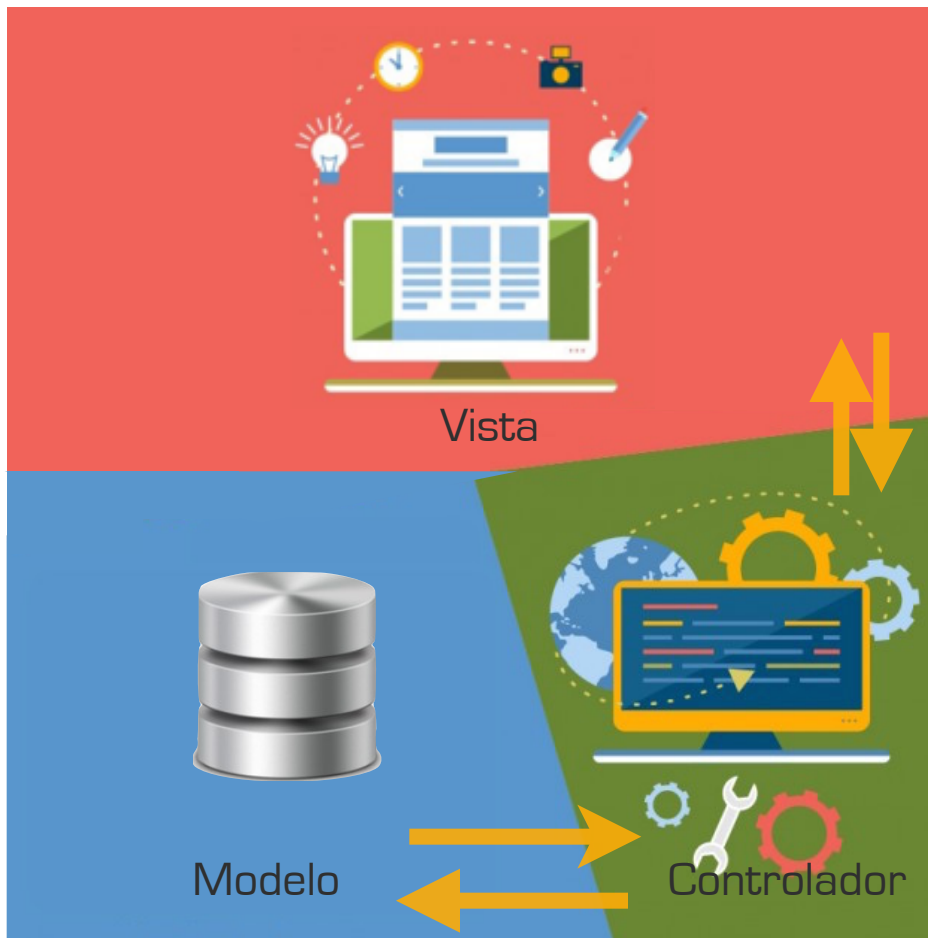


Figura 3.1 Patrón MVC

3.2 Diseño de requisitos funcionales

En esta sección se muestran los requisitos funcionales, representados por medio del diagrama UML de casos de uso, con la documentación respectiva.

En la figura 3.2 se muestra el diagrama de casos de uso referente al panel de control, que es una visión general de los requisitos nominales orientados a las funciones de los usuarios administrador y super-administrador.

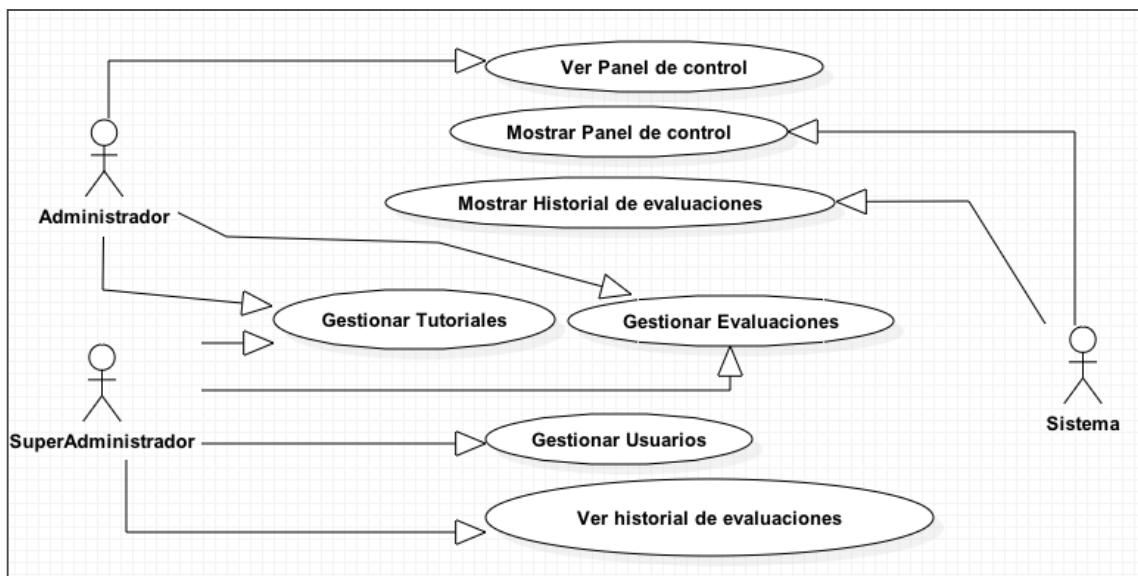


Figura 3.2 Diagrama de casos de uso: Panel de control.

En la tabla 3.1 se presenta la documentación del Caso de uso panel de control.

Tabla 3.1 Documentación de casos de uso: Panel de control	
Nombre	Panel de control
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	El sistema permitirá al administrador gestionar los tutoriales y las evaluaciones, de formas separadas, es decir existirá un módulo para gestionar tutoriales y otro módulo para gestionar exámenes, además existirá un usuario súper administrador que podrá gestionar los mismos casos que el usuario puede administrar y además podrá gestionar alumnos.
Actores	Administrador, Super-administrador, Sistema.
Precondiciones	Estar registrados, y haber iniciado sesión.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema valida el tipo de usuario 2. Despliega el menú correspondiente 3. El administrador elige una opción del menú 4. El sistema despliega al usuario a esa opción.
Flujo Alternativo	4. el usuario regresa al inicio del sistema.
Poscondiciones	Ninguno

A continuación en la figura 3.3 se muestran los requisitos nominales de manera general para los usuarios invitados y alumno.

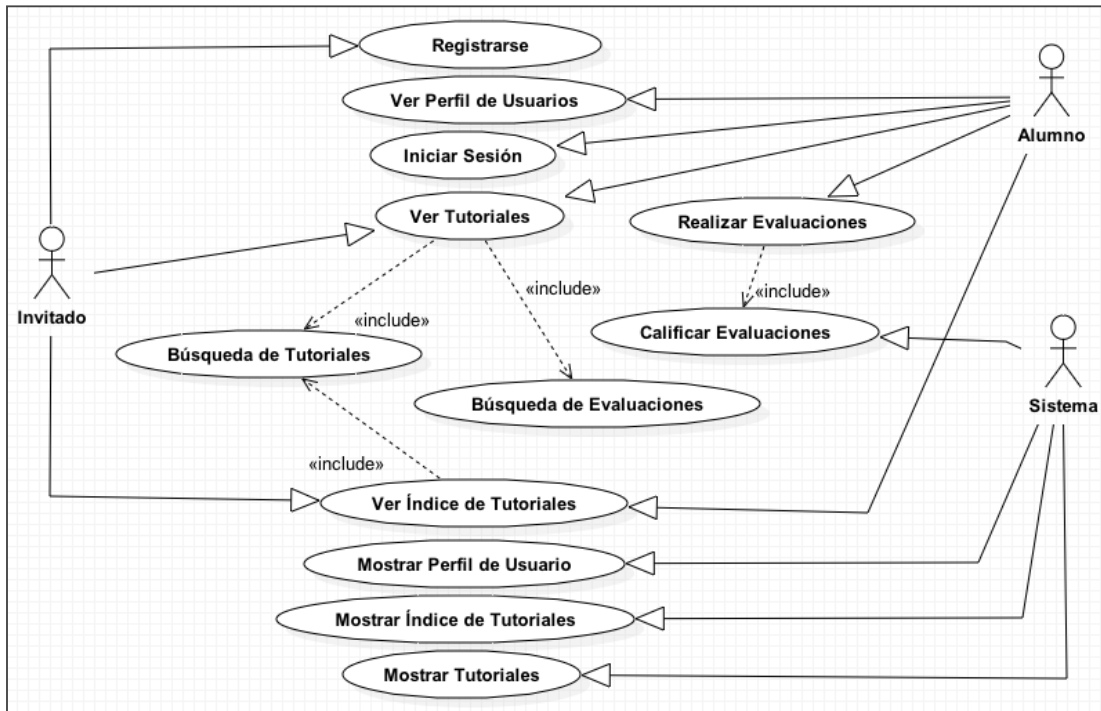


Figura 3.3 Diagrama de casos de uso: General.

En la tabla 3.2 se presenta la documentación de caso de uso General de la figura 3.3

Tabla 3.2 Documentación de casos de uso: general	
Nombre	General
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	El sistema web cuenta con diferentes tipos de usuario, usuarios invitados y alumnos, que de entrada pueden interactuar con distinta información ya que cada usuario tiene distintos privilegios.
Actores	Invitado, Alumno, Sistema
Precondiciones	Ninguna

Flujo normal

1. El Actor invitado puede registrarse
2. El actor invitado ahora es alumno
3. Puede ver su perfil
4. Ver índice y elegir un tutorial
5. Ver el tutorial
6. Realizar una evaluación
7. Cerrar sesión

Flujo Alternativo

1. El actor invitado busca un tutorial
2. Ver el índice de tutoriales
3. ver un tutorial

Poscondiciones Ninguno

En la figura 3.4 se muestra el diagrama de casos de uso del FN.2 Gestión de tutoriales.

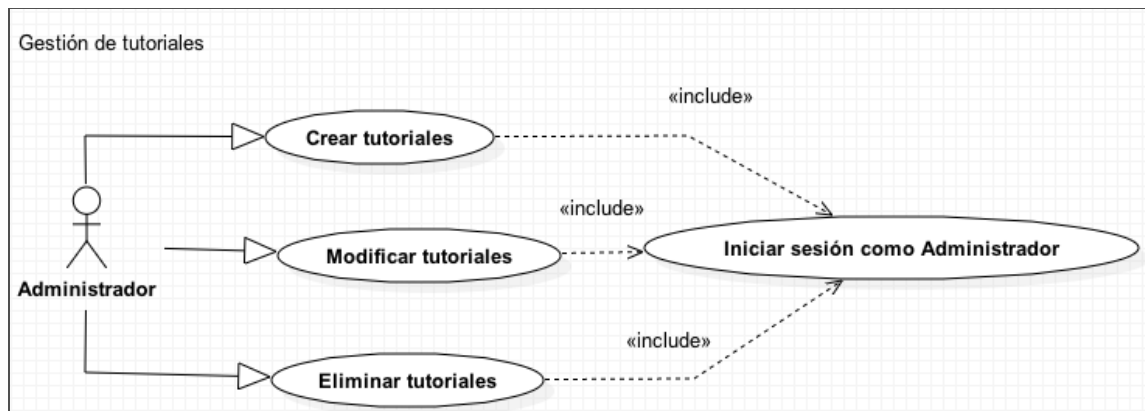


Figura 3.4 Diagrama de casos de uso: Gestión de tutoriales.

En la tabla 3.3 se muestra la documentación respectiva al requisito funcional FN.2.

Tabla 3.3 Documentación de casos de uso: Gestión de tutoriales	
Nombre	Gestión de tutoriales
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	Este requisito permitirá habilitar la creación y modificación de contenido nuevo en cuanto a tutoriales se refiere, estos deberán ser creados a través de un formulario, de la misma forma existirá un apartado para modificar o eliminarlo.
Actores	Administrador
Precondiciones	Haber iniciado sesión como administrador.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrador, debe haber iniciado sesión como administrador. 2. Crea un tutorial para publicarlo. 3. Modifica un tutorial ya existente. 4. Elimina un tutorial ya existente.
Flujo Alternativo	—
Poscondiciones	Ninguno

En el siguiente diagrama de casos de uso, se muestra el FN.3 Gestión de Evaluaciones en la Figura 3.5.

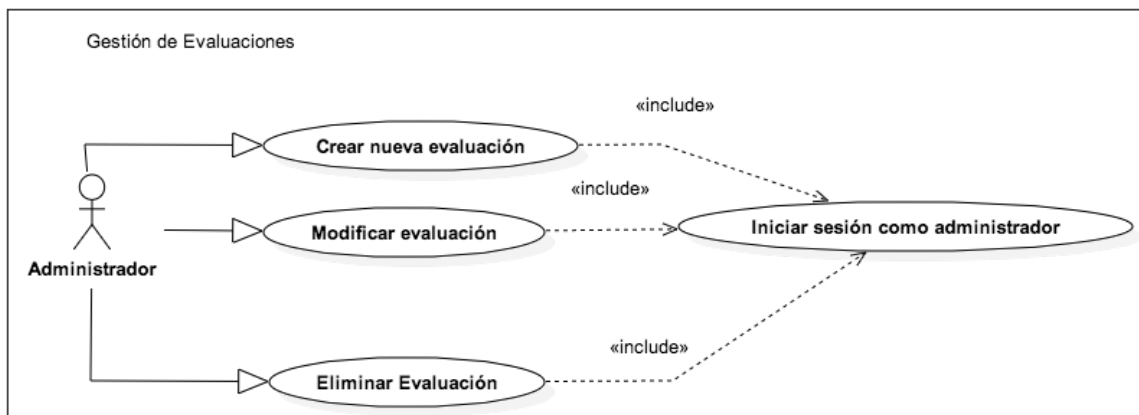


Figura 3.5 Diagrama de casos de uso: Gestión de Evaluaciones

La documentación de este caso de uso se muestra en la tabla 3.4.

Tabla 3.4 Documentación de casos de uso: Gestión de evaluaciones	
Nombre	Gestión de Evaluaciones
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	La gestión de exámenes permite al administrador crear, modificar o eliminar exámenes. Para poder crear exámenes se necesita de un formulario ya que permite crear exámenes de una manera más fácil. Así mismo mediante un formulario permitirá modificar o eliminar los exámenes.
Actores	Administrador
Precondiciones	Haber iniciado sesión como administrador.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrador, debe haber iniciado sesión como administrador. 2. Crea una evaluación nueva 3. Modificar una evaluación existente 4. Eliminar una evaluación existente
Flujo Alternativo	—
Poscondiciones	Ninguno

En la figura 3.6 se muestra el diagrama de casos de uso para el FN.4 Gestión de usuario.

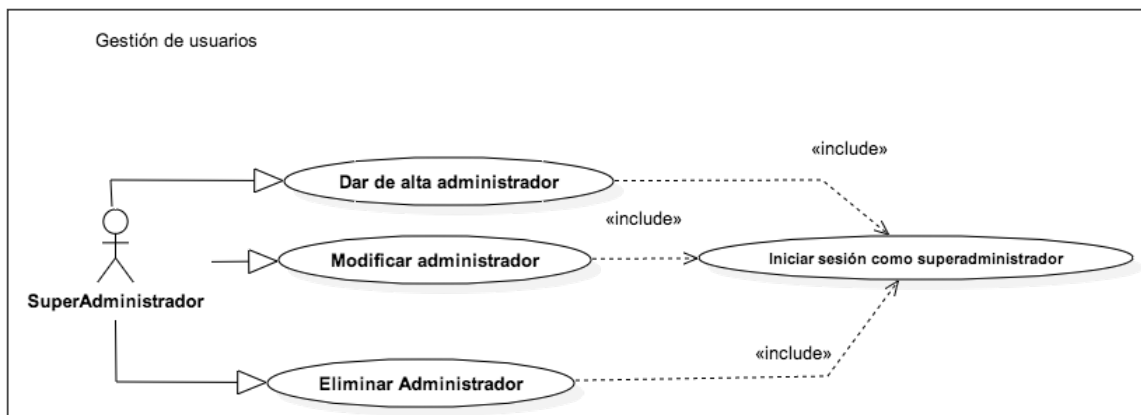


Figura 3.6 Diagrama de casos de uso: Gestión de Usuarios

En la siguiente tabla 3.5 se muestra la documentación del FN.4 gestión de usuarios.

Tabla 3.5 Documentación de casos de uso: Gestión de usuarios	
Nombre	Gestión de usuarios.
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	La gestión de usuarios permite al administrador dar de alta o baja a usuarios de tipo administradores o alumnos, pero esta función solo aplica para un usuario super usuario (los alumnos si pueden darse de alta como usuarios de tipo alumno). Los tipos de usuarios son Administrador, Alumno, e invitado.
Actores	Super-administrador
Precondiciones	Iniciar sesión como super-administrador.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrador, debe haber iniciado sesión como administrador. 2. Escribir su código de super-administrador. 3. Puede crear un usuario nuevo. 4. Modificar un usuarios nuevo. 5. Eliminar un usuario nuevo.
Flujo Alternativo	—
Poscondiciones	Ninguno

En la figura 3.7 se muestra el diagrama correspondiente al requisito funcional nominal FN.5 Historial de alumnos.

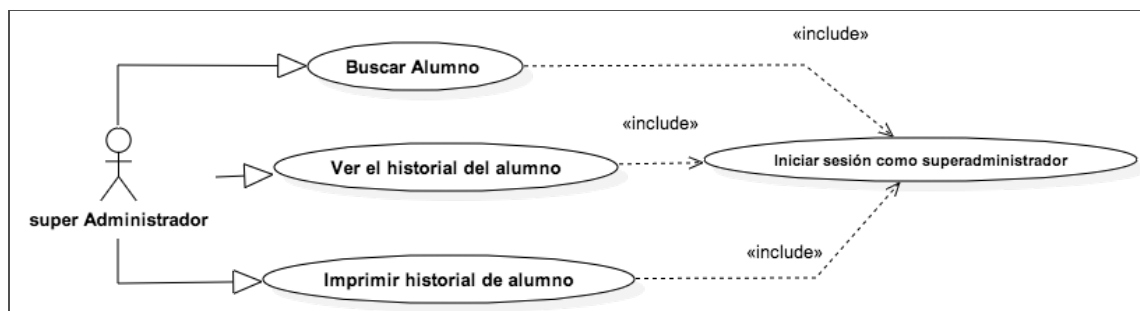


Figura 3.7 Diagrama de casos de uso: Historial de alumnos.

En la tabla 3.6 se muestra la documentación asociada al diagrama de casos de uso Historial de alumnos.

Tabla 3.6 Documentación de casos de uso: historial de alumnos	
Nombre	Historial de alumnos.
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	En el sistema el super usuario puede también revisar el avance de los alumnos, es decir, se podrá consultar el historial de exámenes resueltos por un alumno.
Actores	Super-administrador
Precondiciones	Iniciar sesión como administrador
Flujo normal	
1. El Administrador, debe haber iniciado sesión como administrador.	
2. Escribir su código de super-administrador.	
3. Buscar un alumno.	
4. Ver el historial de alumno.	
Flujo Alternativo 5. Imprimir el historial de alumno.	
Poscondiciones	Ninguno

El siguiente Diagrama de casos de uso se muestra el FN.6 registro de usuarios, en la figura 3.8.

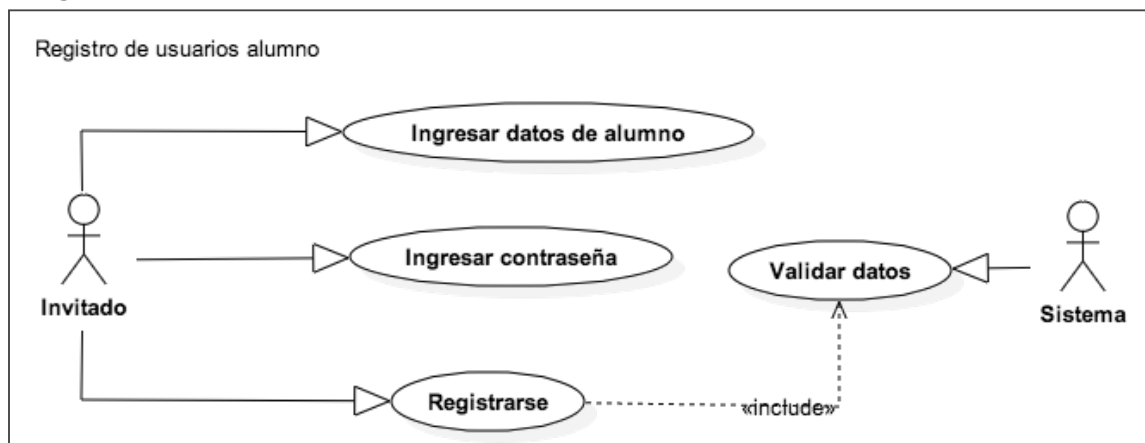


Figura 3.8 Diagrama de casos de uso: Registro de usuarios.

La documentación para el diagrama de casos de uso FN.6 Registro de usuarios, se encuentra en la tabla 3.7.

Tabla 3.7 Documentación de casos de uso: Registrar usuarios	
Nombre	Registrar usuarios.
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	El sistema contará con la capacidad para que usuarios de tipo invitado puedan registrarse como alumnos, así poder realizar evaluaciones de sus conocimientos.
Actores	Invitado
Precondiciones	Ninguna
Flujo normal	
<ol style="list-style-type: none"> 1. El usuario invitado accederá al registro de usuarios. 2. Deberá ingresar sus datos que tiene como alumno. 3. Proporcionará una contraseña para su cuenta de usuario. 4. Para finalizar manda el formulario de registro al sistema para registrarse. 	
Flujo Alternativo 4. Cancelar.	
Poscondiciones	Ninguno

En el diagrama de casos de uso de la figura 3.9 muestra el requisito funcional nominal FN.7.

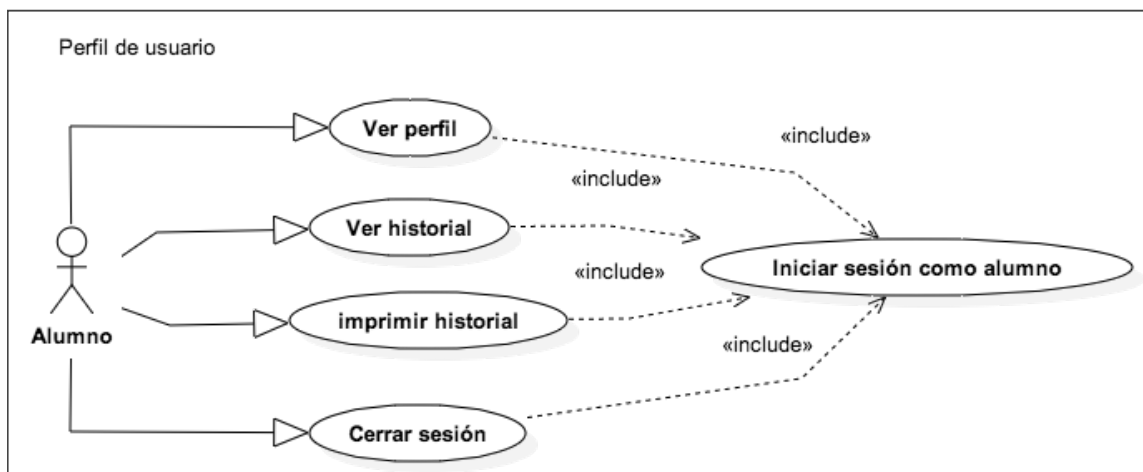


Figura 3.9 Diagrama de casos de uso: Perfil de usuario de tipo alumno.

La documentación respectiva al FN.7 se muestra en la siguiente tabla 3.8

Tabla 3.8 Documentación de casos de uso: Perfil de usuarios de tipo alumno.	
Nombre	Perfil de usuarios de tipo alumno.
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	Una vez que algún usuario invitado se registre y/o inicie sesión, este podrá acceder a su perfil, mismo donde estarán algunos datos como su Historial de calificaciones.
Actores	Alumno
Precondiciones	Iniciar sesión como alumno.
Flujo normal	
1. El usuario invitado deberá haber iniciado sesión como alumno.	
2. Accederá al perfil de usuario.	
3. Visualizará su historial.	
Flujo Alternativo 4. Imprimir historial	
Poscondiciones	Ninguno

El FN.8 se muestra en la siguiente figura 3.10 diagrama de casos de uso para mostrar tutoriales.

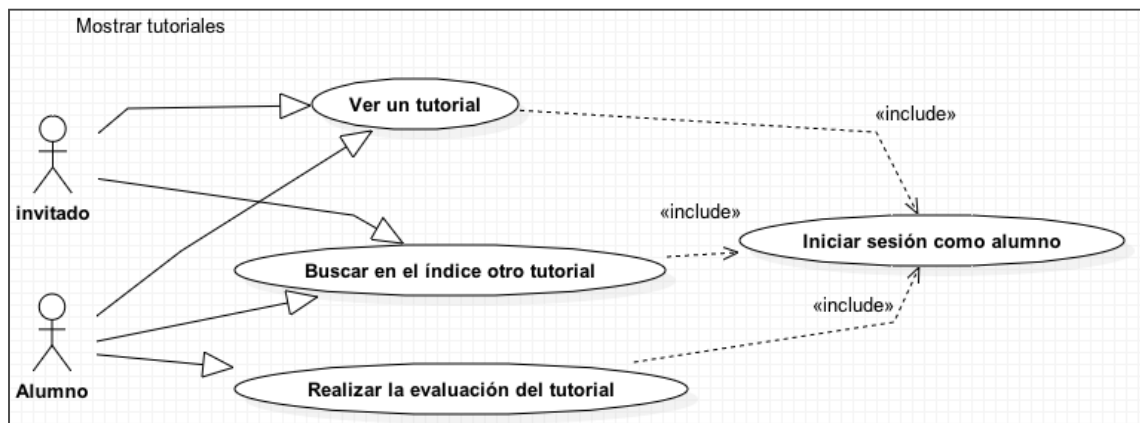


Figura 3.10 Diagrama de casos de uso: Mostrar tutoriales

La documentación para el caso de uso de la figura 3.10 Mostrar tutoriales se encuentra en la siguiente tabla 3.9.

Tabla 3.9 Documentación de casos de uso: Mostrar tutorial	
Nombre	Mostrar tutoriales
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	El sistema cuenta con un módulo para visualizar los tutoriales que existan, cada uno contará con un título, su contenido, imágenes y líneas de código en lenguaje C si se necesitan.
Actores	Alumno, invitado
Precondiciones	Ninguna
Flujo normal	<ol style="list-style-type: none"> 1. El usuario invitado podrá o no haber iniciado sesión como alumno. 2. Puede ver un tutorial 3. Buscar un tutorial.
Flujo Alternativo	4. Sí inició sesión como alumno puede realizar la evaluación respectiva al tutorial que esté visualizando.
Poscondiciones	Ninguno

El FN.9 se muestra en la siguiente figura 3.11 diagrama de casos de uso para Índice de tutoriales.

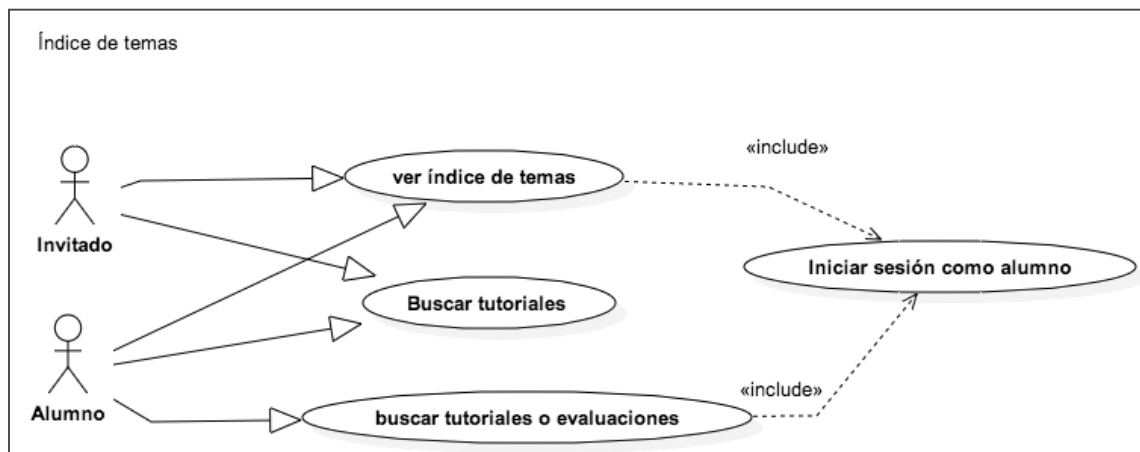


Figura 3.11 Diagrama de casos de uso: Índice de tutoriales.

En la tabla 3.10 se muestra la documentación respectiva para el diagrama de casos de uso del índice de tutoriales.

Tabla 3.10 Documentación de casos de uso: Índice de tutoriales	
Nombre	Índice de tutoriales
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	El sistema tendrá un índice de temas el cual deberá de organizarse de menor a mayor dificultad.
Actores	Alumno, invitado
Precondiciones	Ninguna
Flujo normal	
1. El usuario invitado puede o no haber iniciado sesión como alumno.	
2. Buscar un tutorial.	
3. Ver el listado de tutoriales.	
Flujo Alternativo 4. Sí inició sesión como alumno puede buscar evaluaciones.	
Poscondiciones	Ninguno

El siguiente diagrama de casos de uso para el requisito FN.11 Realizar evaluaciones se muestra en la figura 3.12

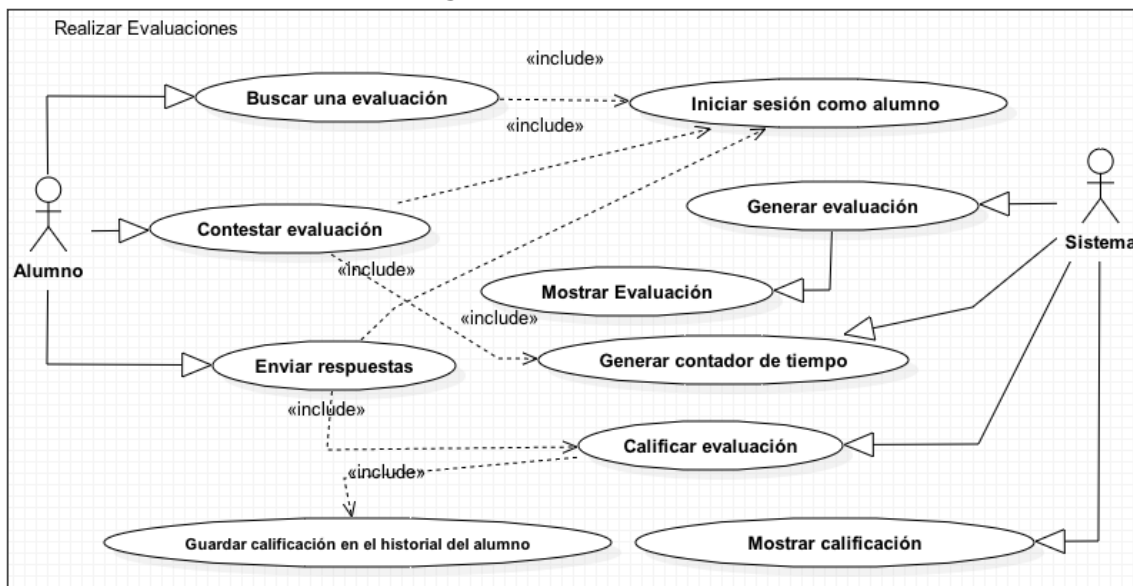


Figura 3.12 Diagrama de casos de uso: Realizar evaluaciones.

La documentación respectiva al diagrama de casos de uso FN.11 Realizar evaluaciones, se encuentra en la tabla 3.11

Tabla 3.11 Documentación de casos de uso: Realizar evaluaciones	
Nombre	Realizar evaluaciones.
Autor	José Manuel Patiño Gutiérrez
Fecha	27 Noviembre 2014
Descripción	El sistema deberá contener un módulo para realizar exámenes de acuerdo al tema que el usuario elija. Deberá tener un tiempo determinado para realizarlo, según la complejidad del tutorial al que corresponda.
Actores	Alumno, Sistema.
Precondiciones	Haber iniciado sesión como alumno.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario habrá iniciado sesión como alumno. 2. El alumno buscará una evaluación. 3. El sistema generará las condiciones para la evaluación es decir: <ol style="list-style-type: none"> 3.1 Generará la evaluación 3.2 Mostrará la evaluación 3.3 Mostrará el tiempo con el que se cuenta para realizar la evaluación. 4. El alumno contesta la evaluación 5. El alumno envía las respuestas al sistema. 6. El sistema califica la evaluación 7. Registra la calificación en el historial del alumno 8. Muestra la calificación al alumno.
Flujo Alternativo	4. Sí inició sesión como alumno puede buscar evaluaciones.
Poscondiciones	Ninguno.

3.3 Diseño de la Interfaz de usuario

La interfaz de usuario ayuda y muestra la maquetación del sistema, lo que permite ver de una manera gráfica, lo que se necesita en el diseño y la estructura de la información, para luego pasarla al código y de esta manera programar de una manera mas rápida y ordenada, también ayuda a generar las hojas de estilo de una manera más rápida ya que la maquetación muestra un poco de la distribución del sistema. Además que ayuda la pre visualización de los datos pueden ser incluidos en la Base de Datos. A continuación se muestran algunas interfaces del sistema.

En la figura 3.13 se muestra la interfaz del sistema para el inicio del sistema.

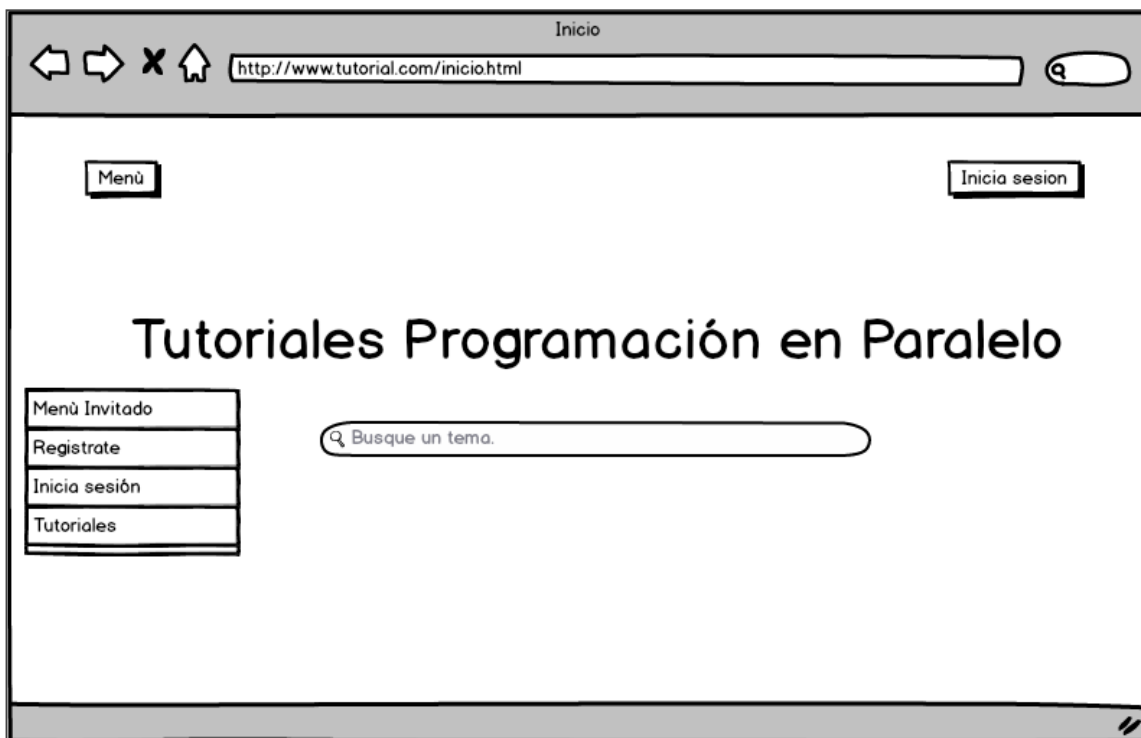


Figura 3.13 Interfaz de la página de inicio del sistema

En la figura 3.14 se muestra la interfaz del sistema dedicada al registro de usuarios de tipo invitado, de tal manera que ya registrados tengan acceso al perfil de usuario de tipo alumno, además les permite realizar evaluaciones y con ello llevar un historial de las evaluaciones realizadas.

The screenshot shows a web browser window with the title 'Registro' and the address bar containing 'http://www.tutorial.com/registrarse/'. The main content area has a heading 'Regístrate' and a paragraph: 'Los usuarios que se registran tienen acceso a un perfil, a realizar evaluaciones, además de llevar un historial de evaluaciones. A continuación proporciona tus datos para registrarte'. Below this is a registration form with the following fields: 'Nombre', 'Apellidos', 'Matricula', 'Grupo', 'Grado', 'Usuario', 'Password', and 'Email'. A 'Registrarse' button is located at the bottom right of the form area.

Figura 3.14 Interfaz del sistema: Registro de usuarios.

En la figura 3.15 se muestra la interfaz del sistema para el inicio de sesión, el inicio de sesión para todos los usuarios (excepto el invitado) es este mismo, el sistema identifica los tipos de usuarios automáticamente mediante su usuario y contraseña. El usuario super administrador cuenta con una clave adicional que ayudará a iniciar sesión con mayor seguridad.

The screenshot shows a web browser window with the title 'Inicia sesion' and the address bar containing 'http://www.tutorial.com/iniciarsesion.html'. The main content area has a heading 'Inicia sesión' and a link '[Aun no estas registrado?](#)' in the top right corner. Below the heading is a login form with the following fields: 'Usuario' and 'Contraseña'. An 'iniciar' button is located at the bottom of the form area.

Figura 3.15 Interfaz del sistema: Inicio de sesión.

En la figura 3.16 Se muestra la interfaz correspondiente al perfil de usuario.

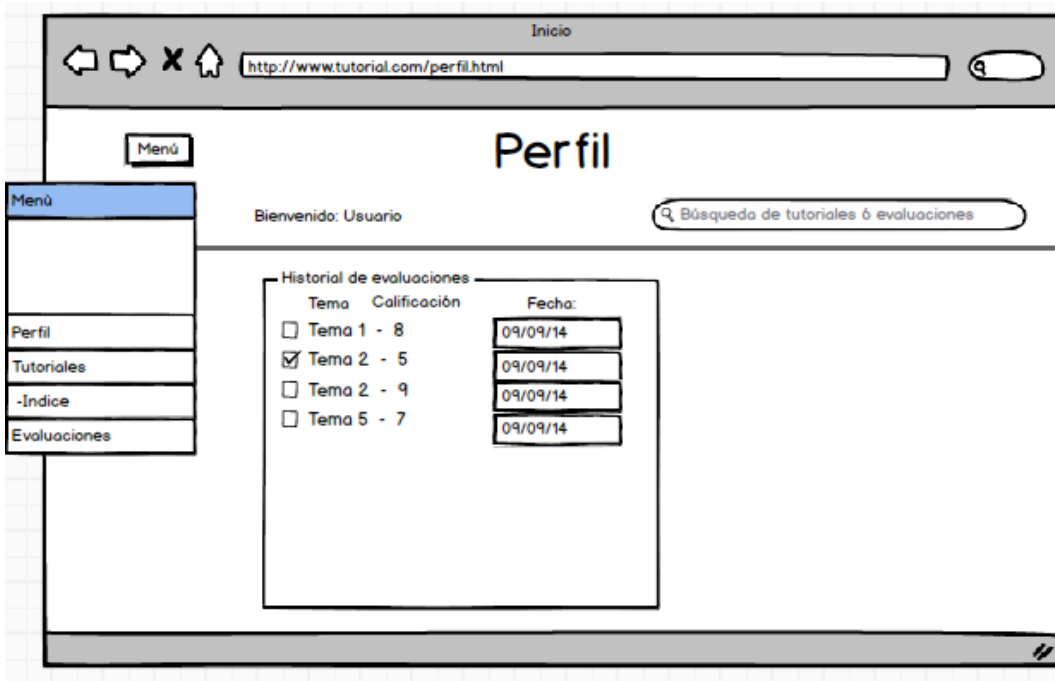


Figura 3.16 Interfaz del sistema: Perfil de usuario de tipo alumno

A continuación se muestra en la figura 3.17 la interfaz de usuario que corresponde al panel de control, misma que puede variar, dependiendo del tipo de usuario, es decir si es super administrador, podrá ver el apartado de gestión de usuarios y el historial de alumnos.

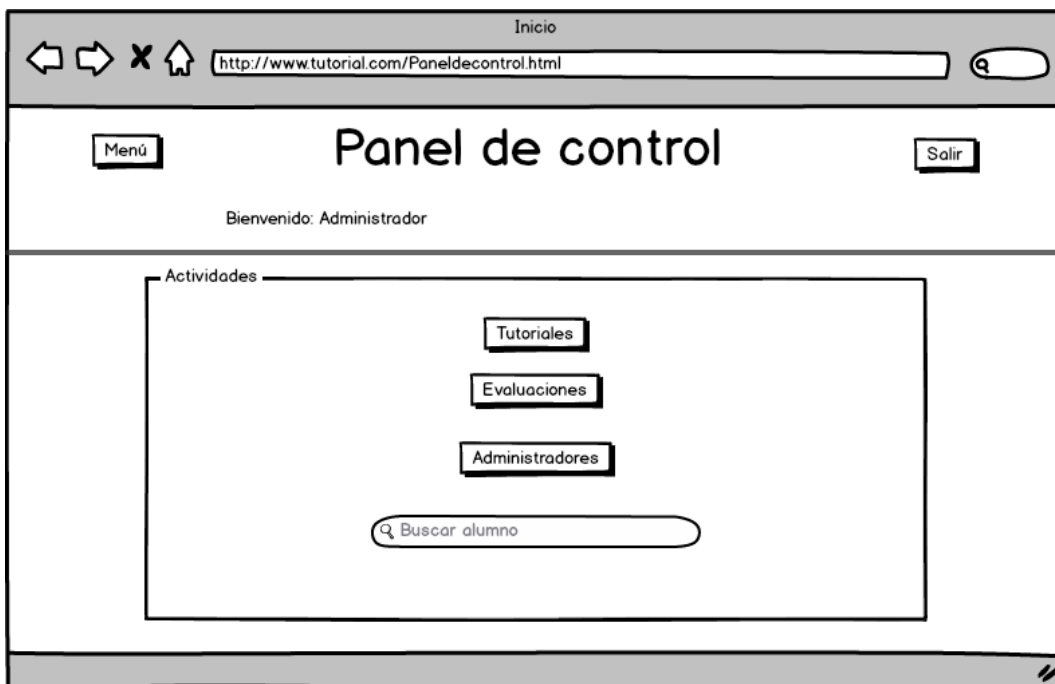


Figura 3.17 Interfaz de usuario: Panel de control.

3.4 Diseño de la Base de datos

Para esta sección se muestra el diagrama E/R de la base de datos así como el diccionario de datos, que ayudarán al sistema a guardar la información que este necesita para funcionar. Una base de datos esta definida por ser como un “almacén” que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente[6].

3.4.1 Diagrama Entidad-Relación

El Dr. Edgar Frank Codd propuso que los sistemas de bases de datos deberían presentarse a los usuarios con una visión de los datos organizados en estructuras llamadas relaciones, definidas como conjuntos de tuplas (filas) y no como series o secuencias de objetos, con lo que el orden no es importante. Por tanto, detrás de una relación puede haber cualquier estructura de datos compleja que permita una respuesta rápida a una variedad de consultas. El Dr. Codd hizo énfasis en que el usuario de un sistema relacional sólo debía preocuparse por el qué consultar y no el cómo de las estructuras de almacenamiento (lo que ahora se conoce como modelo físico). Aún hoy se consideran validas sus afirmaciones a pesar de ser establecidas en 1979[7].

Un diagrama entidad relación está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados entidades, y de relaciones entre estos objetos. Una entidad es una cosa u objeto en el mundo real que es distinguible de otros objetos. El modelado E-R, modela los datos de acuerdo a la manera en que se asocian unos con otros. En el diagrama E-R se tiene la flexibilidad del modelo relacional. En la figura 3.18 se muestra el diagrama entidad relación de este sistema.

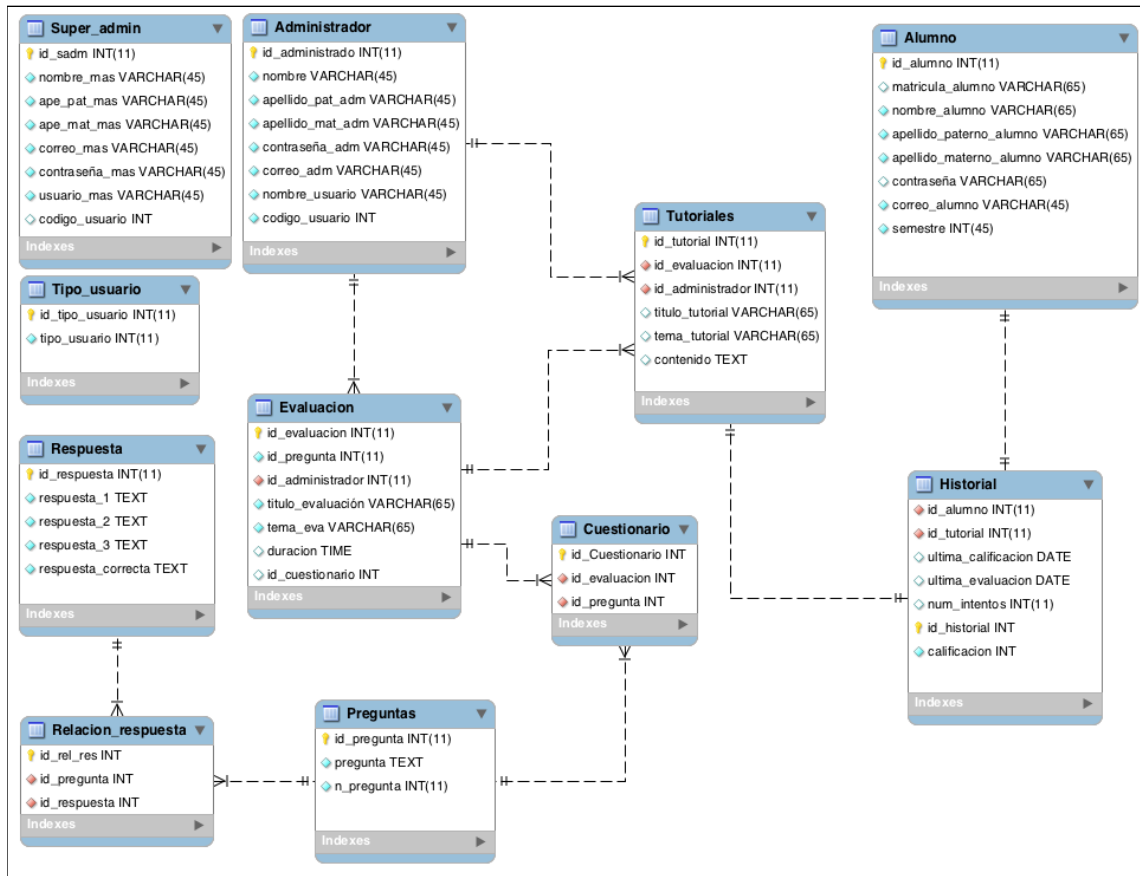


Figura 3.18 Diagrama Entidad-Relación del sistema.

3.4.2 Diccionario de datos.

A continuación se muestra el diccionario de datos del sistema que permite visualizar los datos y descripción que el sistema utiliza, es decir las características y de manera general como se definen, esto ayuda a entender uno por uno los datos que se almacenan.

En las figuras posteriores podemos observar el diccionario de datos correspondiente al sistema donde:

PK: llave primaria

NN: No nulo

AI: Se auto incrementa

En la figura 3.19 se muestra la tabla alumnos:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id_alumno	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único correspondiente al alumno
matricula	VARCHAR(13)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Matrícula correspondiente al alumno'
nombre	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Nombre del alumno'
ape_pat	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Apellido paterno del alumno'
ape_mat	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Apellido materno del alumno'
contraseña	VARCHAR(16)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Contraseña del alumno'
correo	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Correo del alumno'
semestre	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Semestre actual del alumno

Figura 3.19 Tabla alumnos

En la figura 3.20 se muestra la tabla administrador:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id_administrado	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único de la tabla
nombre	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Nombre del usuario administrador'
apellido_pat_adm	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Apellido paterno del usuario administrador'
apellido_mat_adm	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Apellido materno del usuario administrador'
contraseña_adm	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Contraseña del usuario administrador'
correo_adm	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Correo del usuario administrador'
nombre_usuario	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Nombre de usuario administrador'
codigo_usuario	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Código usuario

Figura 3.20 Tabla administrador

En la figura 3.21 se muestra la tabla master, que es para el super administrador:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id_sadm	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
nombre_mas	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Nombre del super-administrador'
ape_pat_mas	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Apellido paterno del super-administrador'
ape_mat_mas	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Apellido materno del super-administrador'
correo_mas	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Correo del super-administrador'
contraseña_mas	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Contraseña del super administrador'
usuario_mas	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Nombre de usuario del super-administrador'
tipo_usuario	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Código de usuario

Figura 3.21 Tabla super-administrador

En la figura 3.22 se muestra la tabla tipo_usuario:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id_tipo_usuario	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único para tipos de usuario
tipo_usuario	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Nombre del tipo de usuario'

Figura 3.22 Tabla tipo_usuario

En la figura 3.23 se muestra la tabla historial:

Column	Datatype		PK	NN	UQ	BIN	UN	ZF	AI	Default
id_historial	INT(11)	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único de la tabla historial
id_alumno	INT(11)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con alumno
ultima_calificacion	DATE	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Útima calificación en fecha
ultima_evaluacion	DATE	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Última evaluación en fecha
num_intentos	INT(11)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Número de intentos en realizar la evaluación
id_evaluacion	INT	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con la evaluación
calificacion	INT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Calificación obtenida

Figura 3.23 Tabla historial

En la figura 3.24 se muestra la tabla tutoriales:

Column	Datatype		PK	NN	UQ	BIN	UN	ZF	AI	Default
id_tutorial	INT(11)	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único de los tutoriales
id_evaluacion	INT(11)	↕	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con la evaluación
id_administrador	INT(11)	↕	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con el administrador
titulo_tutorial	VARCHAR(65)	↕	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Título del tutorial'
tema_tutorial	VARCHAR(65)	↕	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Tema del tutorial'
contenido	TEXT	↕	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Link del contenido del tutorial'

Figura 3.24 Tabla tutoriales

En la figura 3.25 se muestra la tabla evaluación:

Column	Datatype		PK	NN	UQ	BIN	UN	ZF	AI	Default
id_evaluacion	INT(11)	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único de la tabla evaluación
id_cuestionario	INT(11)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con el cuestionario
id_administrador	INT(11)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con el administrador
titulo_evaluación	VARCHAR(65)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Título de la evaluación'
tema_eva	VARCHAR(65)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Tema de la evaluación'
duracion	TIME	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Duración HH:MM de la evaluación

Figura 3.25 Tabla evaluación

En la figura 3.26 se muestra la tabla preguntas:

Column	Datatype		PK	NN	UQ	BIN	UN	ZF	AI	Default
id_pregunta	INT(11)	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único de la pregunta
pregunta	TEXT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Texto de la pregunta'
n_pregunta	INT(11)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Número de la pregunta

Figura 3.26 Tabla preguntas

En la figura 3.27 se muestra la tabla respuesta:

Column	Datatype		PK	NN	UQ	BIN	UN	ZF	AI	Default
id_respuesta	INT	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador único de las respuestas
respuesta_1	VARCHAR(45)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Posible respuesta'
respuesta_correcta	VARCHAR(45)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Respuesta correcta'
respuesta_2	VARCHAR(45)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Posible respuesta'
respuesta_3	VARCHAR(45)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Posible respuesta'

Figura 3.27 Tabla respuesta

En la figura 3.28 se muestra la tabla Cuestionario:

Column	Datatype		PK	NN	UQ	BIN	UN	ZF	AI	Default
id_Cuestionario	INT	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador Único de la tabla Cuestionario
id_evaluacion	INT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con Evaluación
id_pregunta	INT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con Pregunta

Figura 3.28 Tabla Cuestionario

En la figura 3.29 se muestra la tabla Relacion_respuesta:

Column	Datatype		PK	NN	UQ	BIN	UN	ZF	AI	Default
id_rel_res	INT	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Identificador Único de la tabla Relación_respuesta
id_pregunta	INT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con la tabla pregunta
id_respuesta	INT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Relacionado con la tabla respuesta

Figura 3.28 Tabla Relacion_preguntas

Capítulo 4 Implementación del sistema

Este capítulo versa sobre la arquitectura física del sistema explicando las características y justificando su uso, además se enlistan las tecnologías utilizadas para el desarrollo e implantación del sistema mismas que se justifican en el apartado 4.2 Tecnologías empleadas. Los errores y problemas que se tengan durante la implementación del sistema serán documentados en el apartado 4.3 solución de problemas de implementación donde además se muestran las soluciones de dichos problemas. Al final de este capítulo se muestra una extracción de código fuente del sistema.

4.1 Arquitectura física

Como se vio en el capítulo 3.1 Arquitectura del sistema, la arquitectura física del sistema comprende la organización del sistema en relación a sus características y componentes. Para la arquitectura física del sistema se utilizará el patrón arquitectónico de capas, utilizando 3 capas: presentación, negocio y datos. [8]

4.1.1 Capa de presentación

Esta capa hace referencia a lo que se presenta al usuario, es decir esta capa interactúa directamente con los datos de entrada, por eso mismo tiene que presentarse de una manera amigable y delicada para no contener información de más o innecesaria. Esta capa interactúa con la capa de lógica del negocio presentada en el siguiente punto 4.1.2. En esta capa es donde más se encuentran aplicados los conceptos de experiencia de usuario, usabilidad etc., para que el usuario pueda identificar rápidamente los elementos del sistema, y de esta forma el usuario adquiera una mayor atracción hacia el sistema. Esta capa es tomada como el cliente [9].

4.1.2 Capa de negocio.

Esta capa se encarga de resolver los problemas de manera directa, es decir el sistema toma en cuenta las restricciones, reglas y operaciones que debe cumplir para realizar de manera efectiva el trabajo, por ejemplo para operaciones como calcular el resultado de un examen, etc. Lo importante de esta capa reside en que si estas operaciones, tareas o reglas manejan información precisa, y el sistema no las realiza correctamente, entonces este pierde funcionalidad, es por ello que se debe tratar con mucha atención el desarrollo de esta capa. Esta capa se relaciona con la capa de presentación, cuando la capa de negocio recibe peticiones del usuario por medio de la interfaz, también se relaciona con la capa de datos, al realizar alguna tarea que interactúe con los datos de la base de datos del sistema, y si así lo requiere regresar a la capa de presentación para mostrar los resultados.

4.1.3 Capa de datos.

Esta capa de datos se enfoca mayormente en la base de datos, es decir se encarga de realizar las transacciones a la base de datos en relación con la capa de negocio, quien envía peticiones para inserciones, consultas o descargas de información según lo requiera el sistema, dicha información debe ser consistente (es decir que no sea ambigua o no tenga incongruencias), además de ser segura y precisa, de esta manera para evitar estas cuestiones la importancia de esta capa está en el diseño de la base de datos, el gestor de la base de datos(SGBD) así como en los protocolos de seguridad del servidor de base de datos.

A continuación se muestra la figura 4.1 que documenta el diagrama de despliegue del sistema de tutoriales, el cual muestra la arquitectura física del sistema.

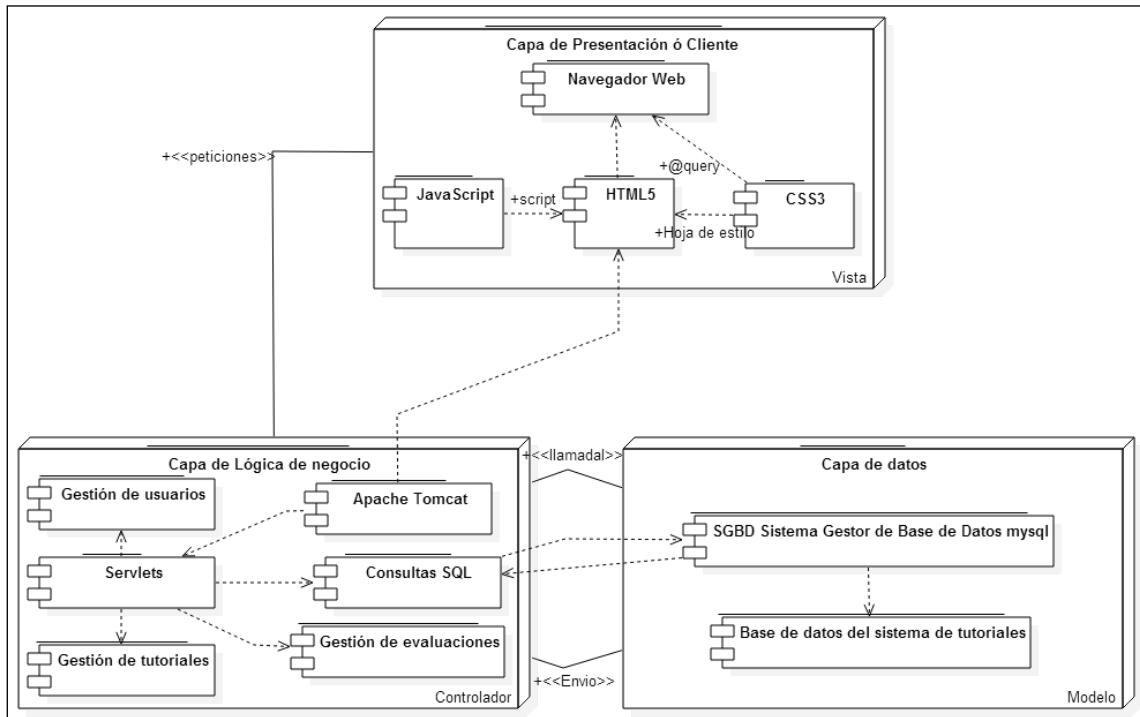


Figura 4.1 Diagrama de despliegue del sistema

4.2 Tecnologías empleadas

Las tecnologías que se relacionan con el desarrollo de sistemas pueden ser muy variadas y de distintos tipos, el uso y manejo de estas mismas, depende mucho del tipo de sistema, en este caso el sistema que es desarrollado y que se presenta en esta tesina, es un sistema Web, es decir que se utilizan tecnologías que a programación web se refieren.

Para el desarrollo de sistemas web existen tecnologías que han quedado obsoletas y con el fin de tener el sistema de una forma actualizada y que permita la escalabilidad según el cliente lo requirió, se hace uso de tecnologías nuevas o de reciente auge. A continuación se listan las tecnologías usadas para el sistema:

- StarUML: StartUML es un software que permite el fácil desarrollo de diagramas principalmente con los estándares UML (Unified Modeling Language ó Lenguaje Unificado de Modelado) y MDA (Model Driven Architecture ó arquitectura dirigida por modelos), se utilizó para cuestiones de diagramas en el capítulo de diseño e implementación del sistema[10].

- J2EE: Java 2 Enterprise Edition (J2EE) es una plataforma de desarrollo en lenguaje java el cual permite utilizar arquitecturas de N capas distribuidas y se ejecuta sobre un servidor de aplicaciones [11].
 - JDK: Es parte de la plataforma de J2EE siendo el Kit de desarrollo en java por sus siglas en inglés Java Developer Kit, que permite la creación de código para el desarrollo de aplicaciones o sistemas.

 - JRE: Al igual que JDK es parte de la plataforma de J2EE siendo por sus siglas en inglés Java Runtime Environment o entorno de ejecución de Java que contiene la máquina virtual de java que pre-compila el código para poder ejecutarlo.

 - JDBC: El conector de base de datos de java por sus siglas en inglés Java Database Connectivity (JDBC), sirve como puente entre el servicio de la base de datos y su utilización en transferencias o consultas, es decir el enlace del sistema como parte del modelo y controlador.

- CentOS: El sistema operativo que se utilizó para los servidores de base de datos y HTTP es CentOS el cual es seguro y confiable, en los últimos años a aumentado su popularidad como sistema operativo para servidor. La versión que se utilizó fue CentOS 6.6 [12].

- Apache Tomcat: Apache Tomcat es el servidor que se utiliza para el dar servicio a las peticiones http además es el contenedor de servlets¹ y Jsp's² de java [13].

¹ Servlet es una conjunto de clases de java que provee las herramientas de comunicación y manejo de peticiones web como el protocolo HTTP.

² Un archivo JSP (Java server pages) permite embeber código java con html, javaScript o csss.

- MySQL: Es el Sistema Gestor de Base de Datos por sus siglas en inglés Database Management System (DBMS) que se utiliza para base de datos y su utilización en modelos relacionales, se utilizó la versión de servidor Mysql-Server [14].
- MySQL Workbench: es una herramienta que permite el diseño de diagramas E/R de bases de datos que además permite la generación de “reverse and forward engineering”, es decir pasar del diagrama al script sql y viceversa [15].
- Eclipse Kepler: Eclipse en su versión Kepler es un IDE (Entorno de Desarrollo Integrado) de desarrollo para Java que se enfoca en desarrollo de sistemas Web utilizando el lenguaje java, provee las herramientas necesarias para la optimización en el desarrollo del mismo [16].
- GitHub: Es una herramienta que permite el control de versiones del proyecto, es decir mantiene los archivos del sistema, registrando los cambios y con la posibilidad de regresar a una versión anterior del sistema [17].
- HTML5: Es el lenguaje de etiquetado utilizado para desarrollo del sistema como parte de la vista, HTML5 es la versión más reciente e incluye nuevos elementos, atributos, y comportamientos que permiten un mejor desarrollo de paginas web. [18].
- CSS3: Es la tecnología de hojas de estilo en cascada, que permiten mediante la interacción con HTML crear reglas y estilos de presentación además de diseño, que mejora la capacidad de mostrar el contenido en el navegador [19].
- JavaScript: El tipo de JavaScript que se utiliza es: Navigator JavaScript y este es un lenguaje para diseño y desarrollo web utilizado para las validaciones, no necesita de un compilador ya que los navegadores se encargan de su interpretación por lo que es un lenguaje de lado del cliente [20].
- JQuery: Es una herramienta basada en JavaScript que facilita interacción con html, eventos, interfaces etc., por medio de una API (Interfaz de Programación de Aplicaciones por sus siglas en inglés) de funciones y librerías de JavaScript, se utiliza en el sistema para realizar apoyo en validaciones y algunos aspectos del diseño[21].

4.3 Solución de problemas de implementación

En este apartado se muestran algunos problemas que se tuvieron mientras se realizaba la solución, mostrando las características de los problemas así como la solución; como en todo desarrollo existen cuestiones que se convierten en retos para los desarrolladores y que tienen que ser resueltos de la mejor manera, para resolver estos problemas es necesario consultar libros, tutoriales, y documentación que oriente al desarrollador a la corrección exitosa.

4.3.1 Botón de menú

En el sistema que se presenta en esta tesina se cuenta con un botón de menú, a diferencia de las páginas web comunes que tiene una barra de navegación estática. Este botón sirve para permitirle al usuario realizar acciones, como ver su perfil, ver el índice, ver tutoriales etc., y el cual despliega un menú solo cuando el usuario requiere verlo, esto ayuda mucho para no distraer al usuario cuando realiza alguna actividad. El problema que se tuvo cuando fue implementada la funcionalidad del menú, fue mostrar el menú con un click para que se deslizara el menú.

Solución:

Se creó una etiqueta de tipo “div”, que contenga el menú que se desea desplegar, como se muestra en las siguientes línea de código:

```
<div id="sidebar">
  <ul>
    <li><a href="Index.jsp"> </a></li>
    <li><a href="Iniciar_sesion.jsp"> Inicia sesión </a></li>
    <li><a href="Indice.jsp"> Índice </a></li>
  </ul>
</div>
```

Después se creó una etiqueta “a” que se utilizó para que este funcione como un botón, de tal manera que con un atributo personalizado “toggle” esté permitida leerse a través de JQuery. Esta función detecta el evento click y ejecuta el desplazamiento por medio de las hojas de estilo (css).

```
<!-- data-toggle para hacer un atributo personalizado a usar con jquery y java script-->
<a href="#" data-toggle=".container" id="sidebar-toggle">
<span class="bar"></span> <span class="bar"></span> <span class="bar"></span></a>
```

código css:


```

#sidebar {
  position: absolute;
  left: -240px;
  background: rgb(59,109,115);
  width: 240px;
  height: 100%;
  box-sizing: border-box;
}

.main-content {
  overflow-x: hidden;
  width: 100%;
  height: 100%;
  padding: 10px;
  box-sizing: border-box;
  -moz-box-sizing: border-box;
  position: relative;
}

.main-content #sidebar-toggle {
  background: rgb(48,84,102); //color
  border-radius: 3px;
  display: block;
  position: relative;
  padding: 10px 7px;
  float: left;
}

/*estilos para la etiqueta span */
.main-content #sidebar-toggle .bar {
  display: block;
  width: 18px;
  margin-bottom: 3px;
  height: 2px;
  background-color: #fff;
  border-radius: 1px;
}

.container.open-sidebar {
  left: 240px;
}

```

Código JavaScript para el botón de menú:

```

//JavaScript del boton de menú utilizando Jquery
$(document).ready(function() {
  //si click en el boton
  $('[data-toggle]').click(function() {
    var toggle_el = $(this).data("toggle");
    //ejecutar el css3 para abrir el menú
    $(toggle_el).toggleClass("open-sidebar");
  });
});

```

En la figura 4.1 se muestra el botón finalizado:



Figura 4.2 Botón para el menú.

4.3.2 Excepción “java.sql.SQLException: No suitable driver found for jdbc”

Con el ID de desarrollo eclipse Kepler a pesar de traer las librerías adecuadas para el desarrollo web, no contiene un controlador específico para la conexión de la base de datos, por lo que en el momento de acceder y hacer peticiones a la base de datos no se encuentra el conector y manda la excepción: “java.sql.SQLException: No suitable driver found for jdbc”

Solución:

La excepción “no suitable driver found for jdbc” se lanza por no encontrar el driver correspondiente para acceder al modelo de datos, es decir a las operaciones con la base de datos del sistema en este caso, para la solución es necesario un controlador que ayude a la interacción entre el lenguaje de programación y el servicio de la base de datos y así evitar que se lance esta excepción. El conector Jdbc se configura de muchas formas, la forma en la que se conectó para el sistema y que se desarrolló fue de la siguiente manera:

- Primero se descarga el driver JDBC de la página oficial: <http://dev.mysql.com/downloads/connector/j/>
- Una vez descargado el driver de la página principal, se procede a abrir eclipse Kepler (el Id de desarrollo) y seleccionar el proyecto, que en este caso es el sistema.

- Posteriormente se va al menú (el cual en la versión en inglés) window->show views->Data source explorer, abrirá la ventana de Data source explorer luego click derecho en Database connections y luego en new como lo muestra la siguiente figura 4.3.

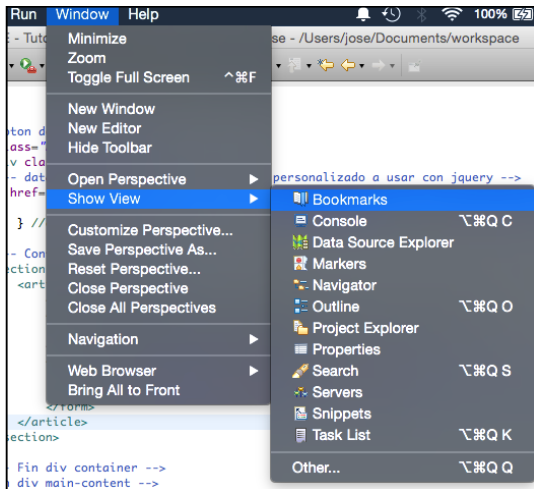


Figura 4.4 Mostrar Data Source Explorer

- Abrirá una ventana donde elegiremos el tipo del conector a configurar, el cual en este caso será mysql agregaremos un nombre a la conexión y posteriormente una descripción, enseguida daremos click en siguiente, como se muestra en la figura 4.5.

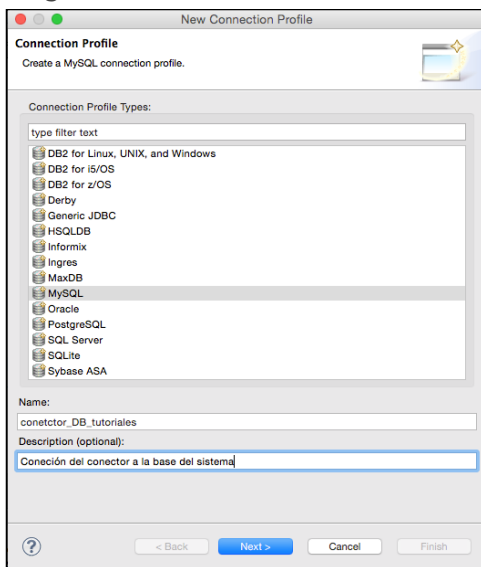


Figura 4.5 Elegir tipo de conector

- En la ventana siguiente en el área de “Drivers” aparece un selector y dos iconos ,en el último icono “Edit Driver Definition” (o editar la definición del driver) daremos click (el botón en forma de triángulo), para configurar el JDBC, como se muestra en la figura 4.6.

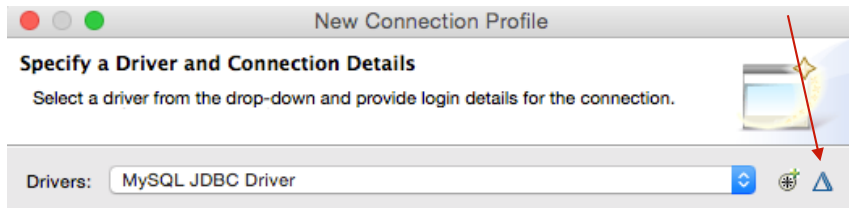


Figura 4.6 Editar la información del driver

- Este abre una nueva ventana con tres pestañas, en la segunda “Jar list” nos permite agregar o editar el path (la dirección donde se encuentra) y posteriormente le damos click en “add jar” y elegimos el archivo que descargamos (el archivo con extensión .jar) en este caso llamado “mysql-connector-java-5.1.34-bin.jar” y lo agregamos, para finalizar damos click en “ok” como podemos observar en la figura 4.7.

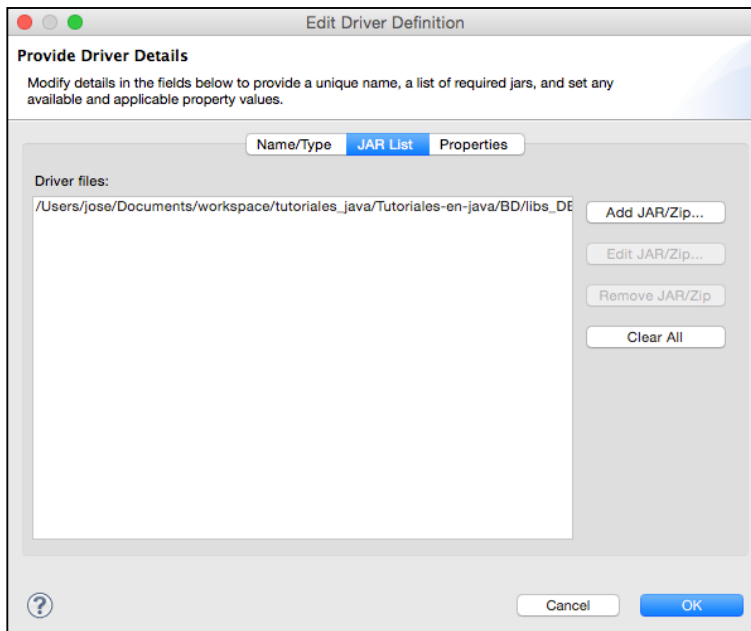


Figura 4.7 Añadir el archivo .jar

- El siguiente paso es configurar el nombre, la URL de la base de datos, y el usuario y contraseña de la base de datos como en la figura 4.8.

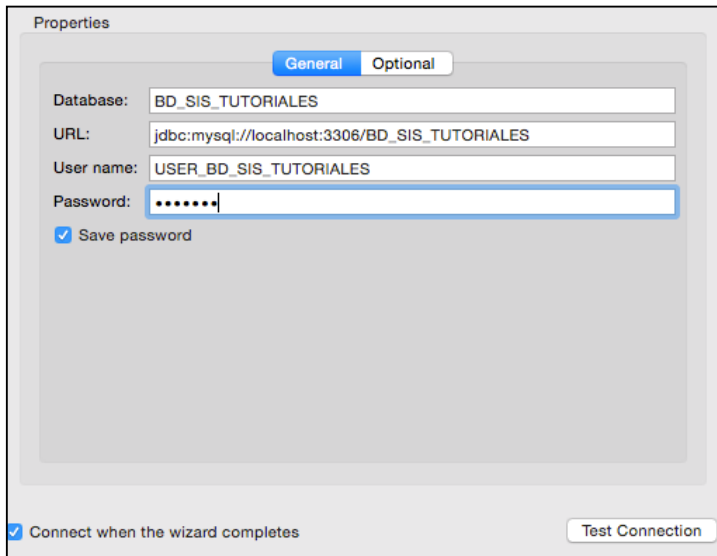


Figura 4.8 Configurar la conexión

- Para finalizar damos click en “test connection” para asegurarnos que todo haya salido bien, si es así nos mostrará el mensaje de “ping succeeded” como se muestra en la figura 4.9 y finalizamos.

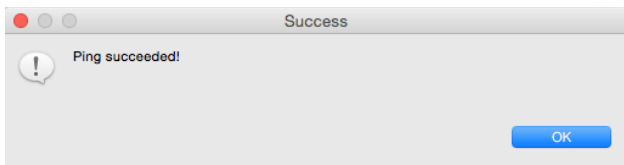


Figura 4.9 Conexión exitosa

4.3.3 Insertar tutoriales.

La inserción de tutoriales es una cuestión que se presenta como un reto debido a que los tutoriales pasan parámetros como imágenes, texto, líneas de código en lenguaje C, el problema está en que no hay una forma en la cual de manera normal se pueda hacer la recepción de estos archivos como las imágenes, por lo que se tiene que utilizar una librería externa y por lo cual cambia por completo la utilización de formularios de html a java.

Solución:

La solución planteada para este problema fue realizarlo usando la librería commons-fileupload-1.3.1, de tal manera que para procesar los formularios html, en java se necesita añadir la librería al path del proyecto y además manejar las peticiones de transacción de información post o get del formulario en un vector, según sea el caso de la información a utilizar. En la documentación de código fuente podremos observar la solución.

4.4 Documentación de código fuente

La documentación del código fuente pretende mostrar la forma en que se realizaron algunas funciones, en este apartado se muestran la documentación del inicio de sesión múltiple y la seguridad de ello, además de la documentación para ver los tutoriales y evaluaciones. La importancia de la documentación de código reside en poder mantener explicada su funcionalidad para con ello ayudar en algún futuro a realizar modificaciones de una manera más fácil y se entienda para que sirve, además también puede funcionar para dar mantenimiento al sistema. Cabe resaltar que tanto la base de datos del sistema, como todos los archivos manejados dentro de este se encuentran con el formato de codificación de caracteres Unicode a 8 bits, es decir “utf-8”[22] que permite el uso de caracteres especiales tales como acentos, letra ñ etc.

4.4.1 Inicio de sesión múltiple

A continuación se muestra la documentación del inicio de sesión. El inicio de sesión es algo particular de este sistema, ya que el mismo formulario permite de manera normal funcionar como inicio de sesión para los usuarios de tipo alumno, pero además a través del mismo inicio de sesión por medio de un código se puede acceder al panel de control, para poder iniciar sesión como usuario de tipo administrador o de tipo master. En la siguiente figura 4.10 se muestra la vista “Iniciar_sesion.jsp” el cual muestra el formulario que requiere de usuario y contraseña para el inicio de sesión.

Al hacer el envío del formulario se realiza mediante el método post, llegando al servlet controlador “Iniciarsesion” donde se obtienen los valores del formulario, se validan los usuarios y posteriormente se crea la sesión, para finalizar con redireccionar al usuario a la respectiva vista, siendo estas el perfil de alumno o el panel de control. A continuación se muestra el código fuente del servlet que verifica el inicio de sesión y el tipo de usuario. Primero mediante un objeto Resultset ³ se obtiene de la base de datos los datos del alumno ó del administrador ó del super administrador para cada caso, se verifica con valor true a la variable “verif” que se muestra a continuación:

```
while(busqueda.next()){  
    //se asigna el valor true si se encuentra el código del tipo de usuario  
    verif=true;  
}
```

“verif” es una variable que después de identificar al tipo de usuario verifica que se inicie la sesión del tipo de usuario, por ejemplo alumno:

```
//si verif es true para alumno:  
if(verif){  
    //Crear la sesion alumno  
    HttpSession usuarios = request.getSession();  
    //asignar variables a la sesión.  
    usuarios.setAttribute("id_alumno",idAlumno);  
    usuarios.setAttribute("nombre", nombre);  
    usuarios.setAttribute("semestre", semestre);  
    usuarios.setAttribute("matricula", usuario);  
    //Dirigir al usuario a su perfil.  
    iniciarsesion_msjs.println("<script> window.location='Perfil';</script>");  
}
```

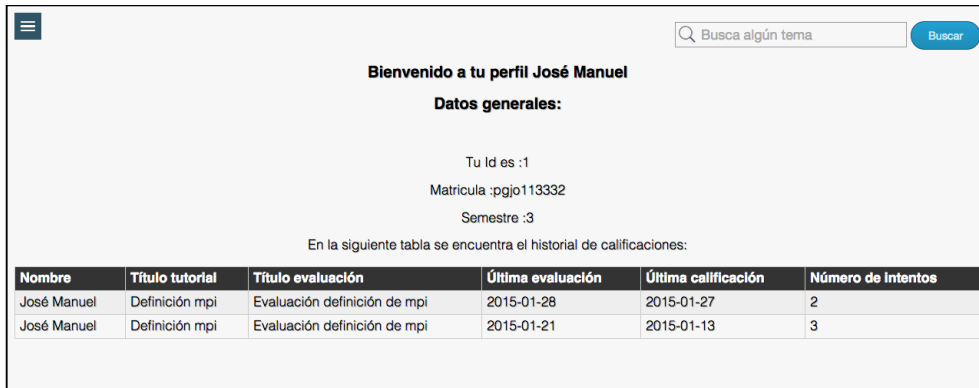
Sí el usuario es administrador en el formulario de Iniciar_sesion.jsp se debe especificar 1 o 2 donde 1 es administrador normal y 2 super-usuario, y un código personal, para después dirigir al usuario al inicio de sesión para el panel de control. A continuación se muestra dicho código fuente.

```
//Sí verif es true para administrador del tipo 1  
if(verif){  
    //se crea la sesión para administrador  
    HttpSession administrador = request.getSession();  
    //se asignan las variables de la sesión  
    administrador.setAttribute("codigo_usuario",codigoUsuario);  
    administrador.setAttribute("tipo","1");  
    //Dirigir al administrador a inicio de sesión para administrador.  
    response.sendRedirect("Panel_control_iniciar_sesion.jsp");  
}
```

³ ResultSet: Es una clase, que hace una abstracción de una tabla de una base de datos, con la cual se puede directamente interactuar entre el código Java y el gestor de base de datos.[23]

4.4.2 Perfil de usuario

El perfil de usuario es donde un usuario de tipo alumno puede revisar sus calificaciones. Para poder acceder a él, el usuario debe estar registrado y debe haber iniciado sesión, el usuario de tipo alumno, al momento de iniciar sesión se le dirige hacia su perfil donde se muestran sus datos y calificaciones, como se muestra en la siguiente figura 4.10.



Bienvenido a tu perfil José Manuel

Datos generales:

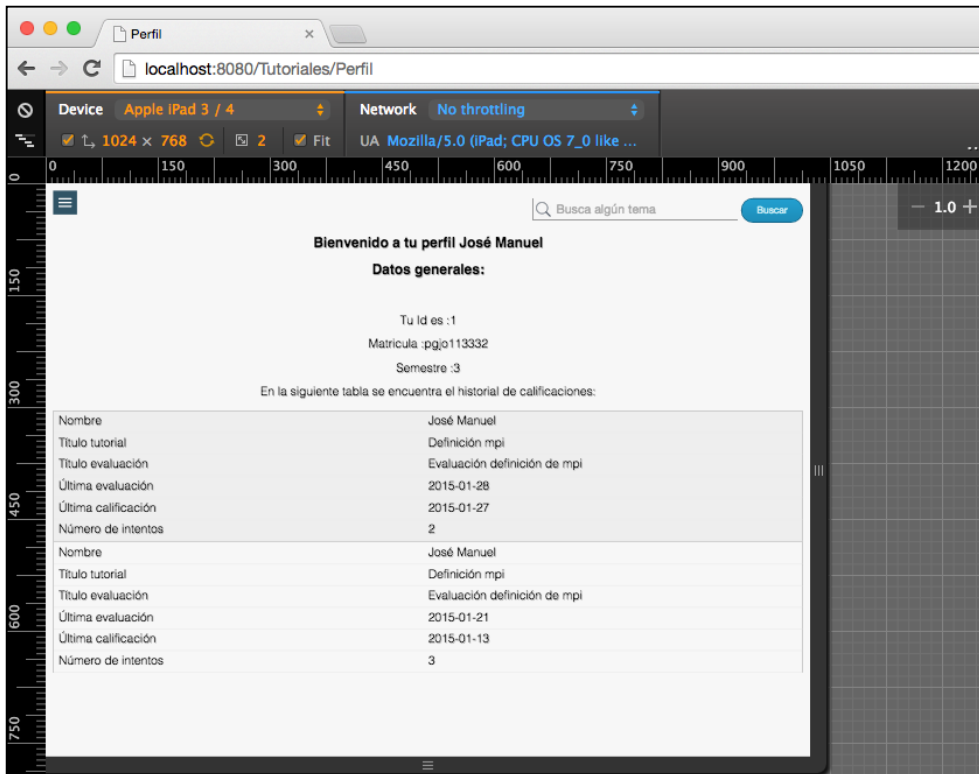
Tu Id es :1
Matricula :pgjo113332
Semestre :3

En la siguiente tabla se encuentra el historial de calificaciones:

Nombre	Título tutorial	Título evaluación	Última evaluación	Última calificación	Número de intentos
José Manuel	Definición mpi	Evaluación definición de mpi	2015-01-28	2015-01-27	2
José Manuel	Definición mpi	Evaluación definición de mpi	2015-01-21	2015-01-13	3

Figura 4.10 Perfil de usuario

Además la tabla de historial de calificaciones, se adapta a la resolución de la pantalla figura 4.11.



Perfil

localhost:8080/Tutoriales/Perfil

Device Apple iPad 3 / 4 Network No throttling

1024 x 768 2 Fit UA Mozilla/5.0 (iPad; CPU OS 7_0 like ...)

Bienvenido a tu perfil José Manuel

Datos generales:

Tu Id es :1
Matricula :pgjo113332
Semestre :3

En la siguiente tabla se encuentra el historial de calificaciones:

Nombre	José Manuel
Título tutorial	Definición mpi
Título evaluación	Evaluación definición de mpi
Última evaluación	2015-01-28
Última calificación	2015-01-27
Número de intentos	2
Nombre	José Manuel
Título tutorial	Definición mpi
Título evaluación	Evaluación definición de mpi
Última evaluación	2015-01-21
Última calificación	2015-01-13
Número de intentos	3

Figura 4.11 Tabla calificaciones con resolución máxima de 1024 por 768 pixeles

A continuación se muestra el código fuente del perfil de usuario:

```
//comprobamos las sesion
HttpSession usuarios = request.getSession();
String nombre = (String)usuarios.getAttribute("nombre");
int idAlumno = (int)usuarios.getAttribute("id_alumno");
int semestre = (int)usuarios.getAttribute("semestre");
String matricula = (String)usuarios.getAttribute("matricula");
//si la sesion no existe redireccionar a cerrar.
if(nombre == null){
    response.sendRedirect("Cerrar");
}

<!-- Mostrar datos del alumno -->);
<h1>Bienvenido a tu perfil "+nombre+" </h1> ";
<h3>Datos generales:</h3><br>";
<p>Tu Id es :"+idAlumno+"</p>";
<p>Matricula :"+matricula+"</p>";
<p>Semestre :"+semestre+"</p>";
<p>En la siguiente tabla se encuentra el historial de calificaciones: </p>";
<!-- Mostrar la tabla de historial -->);
<article>");
    <table>");
        <thead>");
<!-- Cabeceras de la tabla-->");
    <tr>");
        <th>Nombre</th>");
        <th>Título tutorial</th>");
        <th>Título evaluación</th>");
        <th>Última evaluación</th>");
        <th>Última calificación</th>");
        <th>Número de intentos</th>");
    </tr>");
</thead>");
<tbody>");

//recorrer todos los registros
while(rows.next()){
    msjs_perfil.println("                <tr>");
    msjs_perfil.println("                <td>"+nombre+"</td>");
    msjs_perfil.println("                <td>"+rows.getObject("titulo_tutorial")+"</td>");
    msjs_perfil.println("                <td>"+rows.getObject("titulo_evaluación")+"</td>");
    msjs_perfil.println("                <td>"+rows.getObject("ultima_evaluacion")+"</td>");
    msjs_perfil.println("                <td>"+rows.getObject("ultima_calificacion")+"</td>");
    msjs_perfil.println("                <td>"+rows.getObject("num_intentos")+"</td>");
    msjs_perfil.println("            </tr>");
}

```

4.4.3 Panel de control

El panel de control es el inicio para los usuarios de tipo administrador y super-usuario, el panel de control cuenta con un menú de opciones que permiten dirigirse a gestionar tutoriales, evaluaciones, además en el caso del super-usuario le permite gestionar alumnos y administradores.

Para entrar al panel de control los usuarios deben estar dados de alta por el super administrador, también deben tener un código que permita su identificación en el sistema para poder iniciar sesión (este código se les da cuando se registran y es un número), una vez iniciada la sesión, el sistema identifica el tipo de administrador y los dirige hacia el panel de control. En la figura 4.12 se muestra la interfaz del administrador y en la figura 4.13 se muestra la interfaz del super-administrador.



Figura 4.12 Interfaz de usuario administrador

Cuando se pasa el mouse por las imágenes muestra las opciones de crear un nuevo o modificar o eliminar un elemento según corresponda, Tutoriales, Evaluaciones, Administradores y a Usuarios.

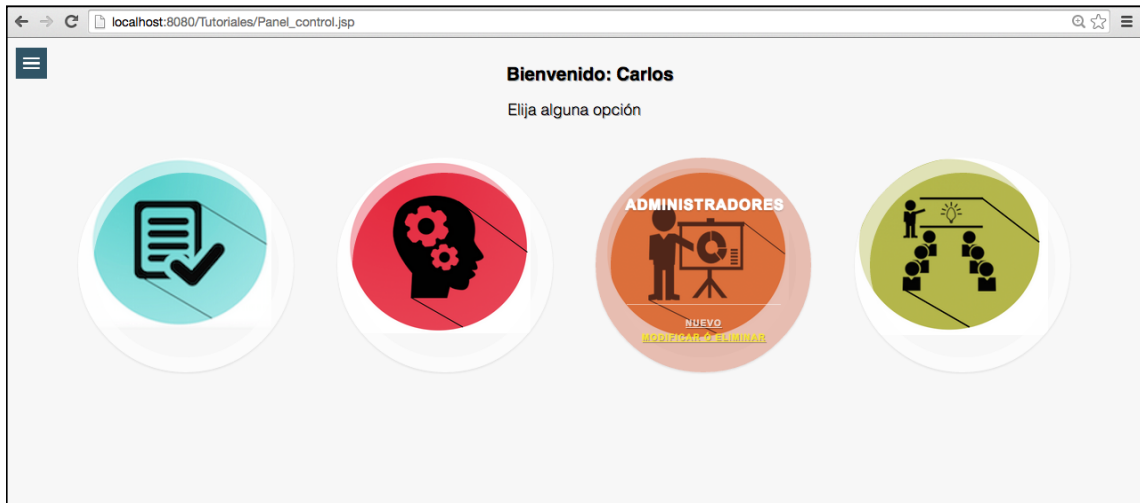


Figura 4.13 Interfaz de usuario super-administrador

A continuación se muestra el código fuente del panel de control:

```
<% //java
    //comprobamos si no hay se sesion de alumno
    HttpSession usuarios = request.getSession();
    String usuario = (String)usuarios.getAttribute("nombre");
    //comprobamos el tipo de administrador
    HttpSession administrador = request.getSession();
    Integer id_administrador = (Integer)administrador.getAttribute("id_administrado");
    String nombre_admin = (String)administrador.getAttribute("nombre_admin");
    String tipo = (String)administrador.getAttribute("tipo");
    //comprobamos si es alumno
    if(usuario == null ){
        //comprobamos si es administrador
        if(!(nombre_admin == null)){
            //si la sesion existe mostrar el menú de opciones
            %>
            <!-- Menú -->
        }
    }
%>
```

Y las opciones según el tipo administrador:

```

<!-- Lista de opciones -->
<!-- Opción 1 -->
<li>
<!-- Añadir los divs ch-item ch-img-1 para especificar imagen opción -->
<div class="ch-item ch-img-1">
<div class="ch-info">
<!-- Texto de la opción 1-->
<h3>Tutoriales</h3>
<p>
<!-- Links para nuevo tutorial y eliminar -->
<a href="Vista_nuevo_tutorial_plantilla.jsp">Nuevo</a>
<a href="vista_modificar_eliminar_tutorial.jsp">Eliminar</a>
</p>
</div>
</div>
</li>
<!-- Opción 2 -->
<li>
<!-- Añadir los divs ch-item ch-img-2 para especificar imagen opción -->
<div class="ch-item ch-img-2">
<div class="ch-info">
<!-- Texto de la opción 2-->
<h3>Evaluaciones</h3>
<p>
<!-- Links para nueva evaluación y eliminar -->
<a href="vista_nuevo_evaluacion.jsp">Nuevo</a>
<a href="vista_modificar_eliminar_evaluacion.jsp">Modificar ó eliminar</a>
</p>
</div>
</div>
</li>
<!-- Opción 3 -->
<li>
<!-- Añadir los divs ch-item ch-img-3 para especificar imagen opción -->
<div class="ch-item ch-img-3">
<div class="ch-info">
<!-- Texto de la opción 3-->
<h3>Administradores</h3>
<p>
<!-- Links para nuevo administrador, modificar y eliminar -->
<a href="vista_nuevo_administrador.jsp">Nuevo</a>
<a href="Vista_administrador_me.jsp">Modificar</a>
<a href="Vista_administrador_e.jsp">Eliminar</a>
</p>
</div>
</div>
</li>
<!-- Opción 4 -->
<li>
<!-- Añadir los divs ch-item ch-img-3 para especificar imagen opción -->
<div class="ch-item ch-img-4">
<div class="ch-info">
<!-- Texto de la opción 3-->
<h3>Alumnos</h3>
<p>
<!-- Links para nuevo alumno gestionar -->
<a href="vista_nuevo_alumno.jsp">Gestionar alumnos</a>
</p>
</div>
</div>
</li>
<%>
//si el usuario es super administrador
if(tipo.equals("2") ){
%>
<!-- Opción 3 -->
<li>
<!-- Añadir los divs ch-item ch-img-3 para especificar imagen opción -->
<div class="ch-item ch-img-3">
<div class="ch-info">
<!-- Texto de la opción 3-->
<h3>Administradores</h3>
<p>
<!-- Links para nuevo administrador, modificar y eliminar -->
<a href="vista_nuevo_administrador.jsp">Nuevo</a>
<a href="Vista_administrador_me.jsp">Modificar</a>
<a href="Vista_administrador_e.jsp">Eliminar</a>
</p>
</div>
</div>
</li>
<!-- Opción 4 -->
<li>
<!-- Añadir los divs ch-item ch-img-3 para especificar imagen opción -->
<div class="ch-item ch-img-4">
<div class="ch-info">
<!-- Texto de la opción 3-->
<h3>Alumnos</h3>
<p>
<!-- Links para nuevo alumno gestionar -->
<a href="vista_nuevo_alumno.jsp">Gestionar alumnos</a>
</p>
</div>
</div>
</li>
<%>
}

```

4.4.4 Añadir tutoriales

En el apartado 4.3.3 se comento la solución planteada referente a la inserción de tutoriales nuevos, en este apartado se muestra el código fuente de la vista que se utiliza para insertar tutoriales, cuyo código fuente se muestra a continuación:

En esta primera vista se da a elegir al usuario que plantilla va a ocupar.

```
<!--Mostrar el nombre del administrador -->
<article>
  <h1>Bienvenido: <%=nombre_admin %></h1>
  <p>Para insertar un tutorial nuevo elija una plantilla:</p><br>
</article>
<!-- Mostrar las plantillas -->
<article id="plantilla">
  <article>
    <!-- Plantilla 1-->
    <a href="vista_nuevo_tutorial.jsp?plantilla=1">
      <!-- Mostrar imagen del formato de la plantilla -->
      
    </a>
    <!-- Pie de página de la plantilla 1 -->
    <p>Plantilla 1: para tutoriales pequeños.</p>
  </article>
  <article>
    <!-- Plantilla 2-->
    <a href="vista_nuevo_tutorial.jsp?plantilla=2">
      <!-- Mostrar imagen del formato de la plantilla -->
      
    </a>
    <!-- Pie de página de la plantilla 2 -->
    <p>Plantilla 2: para tutoriales con utilización de imágenes, código.</p>
  </article>
  <article>
    <!-- Plantilla 3-->
    <a href="vista_nuevo_tutorial.jsp?plantilla=3">
      <!-- Mostrar imagen del formato de la plantilla -->
      
    </a>
    <!-- Pie de página de la plantilla 1 -->
    <p>Plantilla 3: para tutoriales grandes con utilización de imágenes y código.</p>
  </article>
  <!-- Mostrar instrucciones al pie-->
  <article>
    <br>
    <!-- Instrucciones para elegir una plantilla -->
    <h2>Para elegir una plantilla solo de click sobre ella.</h2>
    <br>
    <!-- Fin de las opciones de las plantillas -->
  </article>
</article>
```

Una vez que el usuario elija una plantilla se pasa a una segunda vista donde se procesa su solicitud, desplegando los elementos de la vista, como se muestra en el siguiente código fuente en el cual se muestra el título y la descripción a agregar, es muy importante que el formulario contenga el atributo multipart/form-data. A continuación se muestra la siguiente parte de la vista, en donde se muestra la plantilla deseada por el usuario.

```

<!-- Formulario con atributo multipart/form-data necesario para subir archivos al servidor -->
<form action="Agregar_tutoriales" method="post" accept-charset="UTF-8" enctype="multipart/form-data">
<!-- Marco para los campos genericos -->
<fieldset>
<legend>Tutorial</legend>
<input type="text" name="titulo" placeholder="Escriba el título del tutorial" required><br><br>
<input type="text" name="descripcion" placeholder="Escriba una breve descripción del tutorial" required><br><br>
</fieldset>
<br><br>
<!-- Marco para los campos de las plantillas -->
<fieldset>
<legend>Cuerpo</legend>
<%
//variables de sesion obtener la plantilla
HttpSession tutoriales = request.getSession();
if(plantilla.equals("1")){
tutoriales.setAttribute("plantilla", "1");
}%>
<fieldset>
<input type="text" id="subtitulo1" name="subtitulo1" placeholder="Subtitulo1" ><br>
<textarea name="parrafo1" rows="10" cols="40" placeholder="Escribe texto aquí..."></textarea><br>
</fieldset>
<fieldset>
<input type="text" name="subtitulo2" placeholder="Subtitulo2" ><br>
<textarea name="parrafo2" rows="10" cols="40" placeholder="Escribe texto aquí..."></textarea><br>
<input type="file" id="files" name="files[]"> <br>
</fieldset>
<fieldset>
<input type="text" name="subtitulo3" placeholder="Subtitulo3"><br>
<textarea name="parrafo3" rows="10" cols="40" placeholder="Escribe texto aquí..."></textarea><br>
</fieldset>
<%
}else if(plantilla.equals("2")){

```

Después se procede a pasar al controlador, en donde se obtiene el id del último tutorial para poder darle un nombre consecutivo a los tutoriales; posteriormente se verifica si es un archivo “multipart” lo que implica que este es para carga de archivos, y después se construye una lista en un vector genérico “List items”.

```

//verificar si es multipart para subir archivos
if(!isMultipart){
//No es multipart redireccionar
response.sendRedirect("Vista_nuevo_tutorial_plantilla.jsp");
}else{
//else IsMultipart Manejador de los elementos
FileItemFactory factory = new DiskFileItemFactory();
ServletFileUpload upload = new ServletFileUpload(factory);
//lista de elementos en un vector genérico
List items = null;
try{
items = upload.parseRequest(request);
}catch(FileUploadException e){
e.printStackTrace();
}
//fin try-catch
//crear iterador
Iterator itr = items.iterator();
//recorrer vector

```

Con lo anterior se crea un “iterador” el cual ayuda a saber si hay elementos en el vector, para obtener los valores se divide en dos formas, si es texto o si es un archivo, en el primer caso se tiene que verificar la plantilla usada, si esto no se hace en el orden específico, la información queda incompleta y desordenada; en el siguiente código fuente se muestra el ciclo mientras donde se recorre el vector:

```

//recorrer vector
while(itr.hasNext()){
    FileItem item = (FileItem) itr.next();
    tutorial_msjs.println("<br> tamaño de item: "+item.getSize()+" linea82: <br>");
    //si es formulario
    if(item.isFormField()){
        //obtener nombre del input y valor
        String name = item.getFieldName();
        String value = item.getString();

        if(name.equals("titulo")){
            tutorial_msjs.println("valor titulo: "+value+"<br>");
            titulo = value;
        }//fin titulo

        if(name.equals("descripcion")){
            tutorial_msjs.println("valor descripcion: "+value+"<br>");
            desc=value;
        }//fin Descripción

        if(num_plantilla.equals("1")){
            if(name.equals("subtitulo1")){
                tutorial_msjs.println("subtitulo1: "+value+"<br>");
                sub1=value;
            }//fin subtitulo1
            if(name.equals("parrafo1")){
                tutorial_msjs.println("parrafo1: "+value+"<br>");
                para1=value;
            }//fin parrafo 1
            if(name.equals("subtitulo2")){
                tutorial_msjs.println("subtitulo2: "+value+"<br>");
                sub2=value;
            }//fin subtitulo 2
            if(name.equals("parrafo2")){
                tutorial_msjs.println("parrafo2: "+value+"<br>");
                para2=value;
            }//fin parrafo 2
            if(name.equals("subtitulo3")){
                tutorial_msjs.println("subtitulo3: "+value+"<br>");
                sub3=value;
            }//fin subtitulo 3
            if(name.equals("parrafo3")){
                tutorial_msjs.println("parrafo3: "+value+"<br>");
                para3=value;
            }//fin parrafo 3
        }
    }
}

```

Una vez obtenidos los valores en orden, se termina la primera iteración, sin recoger las imágenes, por lo que en la siguiente iteración se recogerá solo un archivo, seguirá iterando hasta obtener todos los archivos. A continuación se muestra el código fuente para las imágenes.:

```

} else {
    //si es archivo imagen:
    try {
        i++;
        String itemName = id_ultimo_tuto+"_"+i+"_"+item.getName();
        //para almacenar con diferente nombre las imagenes se usó un contador
        if(i == 1){
            intemNameimg= "../img/"+itemName;
        } else if(i == 2){
            intemNameimg2= "../img/"+itemName;
        }
        //crear archivo
        File savedfile = new File (ubicacionarchivo,itemName);
        tutorial_msjs.println("<br>Ubicacion archivo1"+ubicacionarchivo+"nombre de la imagen"+itemName+"<br>");
        //guardar imagen
        item.write(savedfile);
        tutorial_msjs.println("Imagen 1 guardada");
    } catch (Exception e) {
        e.printStackTrace();
    } //fin try image
} //fin if formfield
} //fin while
} //isMultipart

```

El siguiente paso en el controlador es generar los archivos, para ello se hace la escritura de acuerdo a la plantilla elegida. Para la escritura de archivos se utilizó la clase "FileWriter" y se generó de acuerdo a la plantilla un archivo .jsp que se guarda en la ubicación del contenido. A continuación se muestra el código fuente del controlador:

```
//ruta del archivo que se va a crear con extencion html
ubicaciontuto = ubicaciontuto+id_ultimo_tuto+".jsp";
try{
    //Crear instancia de filwriter para crear y escribir en un archivo
    FileWriter fw = new FileWriter(new File(ubicaciontuto),true);
    //cabecera del documento
    fw.write("<!DOCTYPE html>"
        + "<head>"
        + "<!-- cabecera del documento -->"
        + "<meta http-equiv='Content-Type' content='text/html;"
        + "charset=utf-8'>"
        + "<title>"+titulo+"</title>"
        + "<link rel='shortcut icon' href='./img/favicon.png' type='image/png' />"
        + "<link rel='stylesheet' href='tuto.css' type='text/css'>"
        + "</head>"
        + "<div class='main-content'> <section class='principal'>");
    //cabecera del cuerpo
    fw.write("<article> <p>"+desc+"</p> </article>");
    //dependiendo de la plantilla se elige el contenido
```

A continuación se muestra el código de la escritura del archivo .jsp de los tutoriales:

```
fw.write("<article> <p>"+desc+"</p> </article>");
//dependiendo de la plantilla se elige el contenido
if(num_plantilla.equals("1")){
    fw.write("<article> <h1 id='sub1'>"+sub1+"</h1><br><p id='para1'>"+para1+"</p><br>");
    fw.write("<h1 id='sub2'>"+sub2+"</h1><br><p id='para2'>"+para2+"</p><br>");
    fw.write("<img src='"+intemNameimg+"' height='15%' width='30%'>");
    fw.write("<h1 id='sub3'>"+sub3+"</h1><br><p id='para3'>"+para3+"</p><br> </article>");
} else if(num_plantilla.equals("2")){
    fw.write("<article> <h1 id='sub1'>"+sub1+"</h1><br><p id='para1'>"+para1+"</p><br>");
    fw.write("<img src='"+intemNameimg+"' height='15%' width='30%'>");
    fw.write("<h1 id='sub2'>"+sub2+"</h1><br><p id='para2'>"+para2+"</p><br>");
    fw.write("<img src='"+intemNameimg2+"' height='15%' width='30%'>");
    fw.write("<div class='code'><pre><code>"+code1+"</code></pre></div>");
    fw.write("<br><p id='para3'>"+para3+"</p><br>");
    fw.write("<h1 id='sub3'>"+sub3+"</h1><br><p id='para4'>"+para4+"</p><br> </article>");
} else if(num_plantilla.equals("3")){
    fw.write("<article><h1 id='sub1'>"+sub1+"</h1>"
        + "<p id='para1'>"+para1+"</p>"
        + "<h1 id='sub2'>"+sub2+"</h1>"
        + "<p id='para2'>"+para2+"</p>"
        + "<div class='code'><pre><code>"+code1+"</code></pre></div>"
        + "<img src='"+intemNameimg+"' height='15%' width='30%'>"
        + "<p id='para3'>"+para3+"</p>"
        + "<div class='code'><pre><code>"+code2+"</code></pre></div>"
        + "<p id='para4'>"+para4+"</p>"
        + "<h1 id='sub3'>"+sub3+"</h1>"
        + "<p id='para5'>"+para5+"</p> </article>");
} //fin plantillas
//pie de pagina
fw.write("</section></div></body> </html>");
//cerrar filewriter
fw.close();
```


Por último se agrega a la base de datos mediante el controlador MySQLDriver de java. En la figura 4.14 se muestran las plantillas del sistema para los tutoriales:

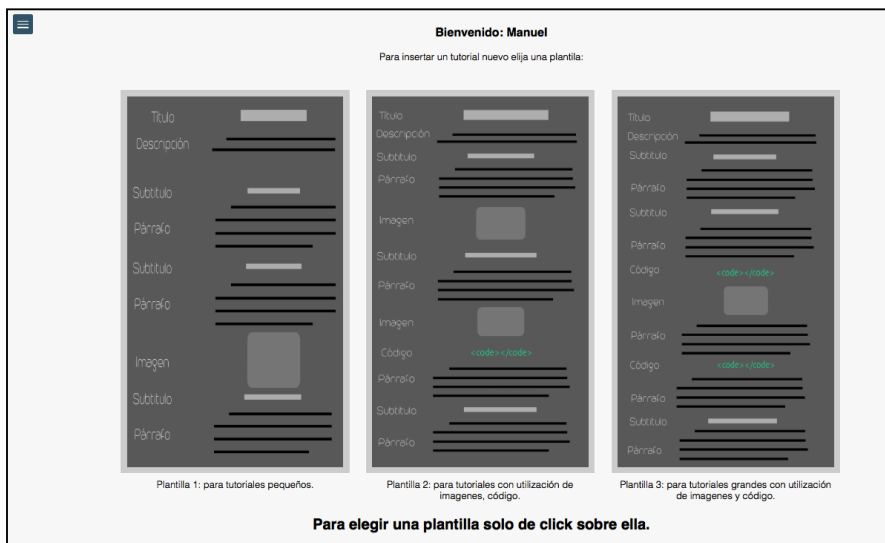


Figura 4.14 Plantillas para los tutoriales.

En la figura 4.15 se muestra el formulario de el tutorial en la plantilla:

The form is titled 'Bienvenido: Manuel' and contains the following elements: a header with a menu icon; a sub-header 'Bienvenido: Manuel'; a note 'Para insertar un tutorial, llene los siguientes campos: *todos los campos son obligatorios'; instructions 'Para agregar elementos utilice los botones' and 'Para finalizar de click en guardar'; a link 'Antes de empezar puede cambiar de plantilla AQUÍ'; a section header 'Tutorial' with two input fields: 'Escriba el título del tutorial' and 'Escriba una breve descripción del tutorial'; a section header 'Cuerpo' with a text area 'Subtitulo1' containing the placeholder 'Escribe texto aquí...'; a file selection button 'Seleccionar archivo' and a text field 'Ningún arch...leccionado'; and a final field 'Subtitulo2'.

Figura 4.15 Formulario del tutorial plantilla 2.

Capítulo 5 Pruebas e implantación

En este capítulo se muestran el ambiente de pruebas, la base de datos de pruebas, la ejecución del plan de pruebas y la bitácora de implantación; el conjunto de todas las pruebas ayudan a clarificar los problemas que se tienen, con el objetivo de encontrar posibles errores, de cualquier tipo, poniendo al sistema a interactuar por primera vez con usuarios, de tal manera que al final de las pruebas pueda implantarse. Esto ayuda a mejorar la calidad y a conocer las opiniones del usuario, además de validar que el sistema funcione correctamente.

5.1 Ambiente de pruebas

Como ambiente de pruebas se utilizarán dos equipos de computo, con distintas características. El ambiente de pruebas tiene como objetivo describir las características de dichos equipos de computo, así como todos los factores que influyen para la realización de las pruebas.

El sistema que se presenta en esta tesina, requiere un navegador web para poder utilizarse, por lo que la gran variedad de navegadores que hay, influyen en el desempeño del sistema, debido a que cada uno de manera interna cuenta con un “motor de renderizado” que interpreta el HTML, el CSS y el JavaScript, puede que no todas las funciones sean compatibles o interpretadas de la misma manera por lo que para este sistema, está principalmente optimizado para ser usado en los navegadores de Google chrome, Opera y Vivaldi sin ningún problema. En la siguiente tabla 9 se muestra las especificaciones técnicas de las computadoras utilizadas como ambiente de pruebas.

Tabla 9. Especificaciones técnicas

Características	Equipo 1	Equipo 2
Modelo	Genérico	Macbook pro
Procesador	Intel Celeron	Intel core i5
Sistema operativo	Windows 7 Home Premium	OS x Yosemite
ID de desarrollo	Eclipse Kepler	Eclipse kepler
Versión del ID de desarrollo	Release 2	Release 2
versión JDK	1.8	1.8
versión JRE	1.8.0_25	1.8.0_20
versión de Tomcat	V. 7.0	V.7.0
Versión del SGBD	4.2.7.1	4.2.7.1
Navegadores utilizados	Google Chrome, Opera	Safari, Vivaldi

A continuación en la tabla 10 se muestran las características de los navegadores utilizados.

Tabla 10. Características de los navegadores

Navegadores	Google chrome	Safari	Opera	Vivaldi
Versión	40.0.2214	8.0.3	27.0.1689.66	1.0.94.2
M o t o r d e navegador web	Blink	web-kit	Blink	Blink
Soporte css	si	si	si	si
S o p o r t e javaScript	si	si	si	si

5.2 Base de datos de pruebas

A continuación se muestran las tablas de la base de datos, estas mismas contienen información que se insertó por medio del sistema excepto en la tabla de super-administrador quien no se agregó por medio del sistema, sino directamente en la base de datos por medio del SGBD.

En la figura 5.1 se muestra la tabla de la base de datos super-administrador.

id_sadm	nombre_mas	ape_pat_mas	ape_mat_mas	correo_mas	contraseña_mas	usuario_mas	codigo_usuario
1	Manuel	Patiño	Gutierrez	patinogutierrez@icloud.com	QWERTY	Usuario1	1910
2	Marco Antonio	Cruz	Chavez	mcruz@uaem.com	mcruz	Dr. Marco	5123

Figura 5.1 Tabla super-administrador

En la figura 5.2 se muestra la tabla de la base de datos administrador.

id_administrado	nombre	apellido_pat_adm	apellido_mat_adm	contraseña_adm	correo_adm	nombre_usuario	codigo_usuario
1	Javier Ignacio	Molina	Cano	12qwasz	javier123@hotmail.com	Javier_Ignacio	7863
2	Lilian Eugenia	Gomez	Alvarez	lili12	lili_gom@icloud.com	lili_12	4043
3	Gerardo Emilio	Duque	Gutierrez	gera12	Gerar_edg@gmail.com	Gera_edg@icloud.com	8641
4	Jose Manuel	Patino	Gutierrez	QWERTY	patinogutierrez@icloud.com	Usuario1	6163
5	Flor Magali	Rodriguez	Soto	magui	fmagalisoto@gmail.com	Magui	3245

Figura 5.2 Tabla administrador

En la figura 5.3 se muestra la tabla de la base de datos tutoriales.

id_tutorial	id_evaluacion	id_administrador	título_tutorial	tema_tutorial	contenido
1	1	1	Que es MPI	Se muestra la introducción de mpi, y distintas car...	./Contenido/tutoriales/1
2	2	1	Comunicación bloqueante	Comunicación bloqueante, funciones y caracteristic...	./Contenido/tutoriales/2
3	5	1	Comunicación punto a punto	Se establecen las definiciones de comunicación pun...	./Contenido/tutoriales/3

Figura 5.3 Tabla tutoriales

En la figura 5.4 se muestra la tabla de la base de datos evaluaciones.

id_evaluacion	id_cuestionario	id_administrador	título_evaluacion	tema_eva	duracion	contenido
1	1	1	¿Que es MPI?	Introducción a MPI	01:20:00	./contenido/evaluacion/id1
2	2	1	Comunicación bloqueante	Evaluar los conocimientos sobre la comunicación bl...	01:00:00	./contenido/evaluacion/id2

Figura 5.4 Tabla evaluaciones

En la figura 5.5 se muestra la tabla de la base de datos preguntas.

id_pregunta	pregunta	num_pregunta
1	¿Que palabras forman las iniciales de MPI?	1
2	¿Cuales son las características de MPI?	2
3	¿En que año fue acabado y publicado el primer est...	3
4	¿Que instrucción se utiliza para compilar MPI?	4
5	¿Como se identifica el comunicador de procesos por...	5
6	¿Cuando usar MPI?	6
7	¿Cuando NO usar MPI?	7
8	¿Cual es el comando para compilar en MPI?	8
9	¿Cual es el comando para ejecutar en MPI?	9
10	¿Que tipo de comunicación ay en MPI?	10
11	¿La comunicación bloqueante esta relacionada con?	1
12	¿Si un send es bloqueante debe también?	2
13	¿Un send o receive bloqueante suspende la ejecució...	3
14	¿Es una característica de las blocking calls?	4
15	¿Cuantas funciones de envío (send) bloqueante son...	5
16	la tarea que hace el envío le indica al receptor ...	6
17	El requiere que haya llegado una notificaciÃ³n di...	7
18	Copia los datos desde el buffer de mensajes a un b...	8
19	El mensaje será copiado del buffer del sistema a l...	9
20	¿La comunicaci3n bloqueante se dice que es?	10

Figura 5.5 Tabla preguntas

En la figura 5.6 y 5.6.1 se muestra la tabla de la base de datos respuestas

id_respuesta	respuesta_1	respuesta_2
1	Mensajería Parcial Interna	Missisipi Programming Institute
2	Servicio, escalabilidad, presentación.	Multiparadigma, multilinguaje, multiprocesador
3	1996	1997
4	#include "mpich.h"	#include "MPI.h"
5	MPI_Comm_rank (MPI_Comm comm , int *rank)	MPI_SEND(MPI_Comm comm, int *rank)
6	En el desarrollo de clases y metodos.	En la implementación de sistemas ERP
7	Si la portabilidad es necesaria.	En el desarrollo de clases y metodos
8	mpireconfig Makefile -np 2	mpire-compile archivo name
9	mpicc -o holamundo holamundo.c	mpirun help
10	síncrono, estándar, write	MPI_COMM_WORLD
11	Blocking calls	Blocking
12	Debe corresponder con un receive no bloqueante	Obligatoriamente un send buffer
13	Un send o receive no bloqueante inicia la comunica...	Un send o receive no bloqueante termina la comunic...
14	Es la forma más segura y la más portable	Termina la comunicación
15	3	5
16	Blocking Ready Send	Blocking Buffered Send
17	Blocking Synchronous Send	Blocking Buffered Send
18	Blocking Ready Send	Blocking Synchronous Send
19	Blocking Buffered Send	Blocking Ready Send
20	Se dice que es una operación de mucha demanda en e...	Se dice que se usa para evitar comunicaciones no-b...

Figura 5.6 Tabla respuestas parte 1

respuesta_3	respuesta_correcta
More Programming Increment	Message Passing Interface
Rapido, multiplataforma, hilos, estructurado.	Portabilidad: multiprocesadores, multicomputadores...
1995	1994
#include "MPICH.h"	#include "mpi.h"
MPI_Comm_size	MPI_COMM_WORD
Es posible usar una biblioteca de más alto nivel	La interrelación de los datos es dinámica o irregu...
La interrelación de los datos es dinámica o irregu...	Es posible usar una biblioteca de más alto nivel
mpirun -help	mpicc -o holamundo holamundo.c
mpireconfig Makefile	mpirun -np 2 -machinefile maquinas holamundo
MPI_barrier, MPI_bcast, MPI_scatter	síncrono, buffer, estándar, ready, recepción
Non-blocking	Las llamadas de send y receive
Debe hacer un send no bloqueante	Debe corresponder con un receive bloqueante
Tiene un impacto en los requerimientos de programa...	El buffer a ser enviado/recibido sea seguro para u...
inicia la comunicación	Pueden estar en correspondencia con non-blocking c...
6	4
Blocking Standard Send	Blocking Synchronous Send
Blocking Standard Send	Blocking Ready Send
Blocking Standard Send	Blocking Buffered Send
Blocking Synchronous Send	Blocking Standard Send
Se dice que es mas eficiente que la comunicación n...	Se dice que es una operación no-local (non-local) ...

Figura 5.6.1 Tabla respuestas parte 2

En la figura 5.7 se muestra la tabla de la base de datos alumnos

id_alumno	matricula_alumno	nombre_alumno	apellido_paterno_alumno	apellido_materno_alumno	contraseña	correo_alumno	semestre
1	GELU150211	German Antonio	Lotero	Upegui	gelu12	Ger_upegui@hotmail.com	1
2	HGCA150111	Hector Ivan	Gonzales	Castañeda	hg12	hect_cast@yahoo.com	4
3	BELT140612	Beatriz Elena	Osorio	Laverde	be12	Bety_lo@yahoo.com	2
4	JUBT150312	Jorge	Uribe	Botero	jorge_ub12	jorge_ub12@gmail.com	1
5	MBLP120215	Maria Isable	Benitez	López	Mari_lo12	Mari_lo@yahoo.com	2
6	lupe	Lizeth	patino	gutierrez	lupe	ligu_12@hotmail.com	2
7	20131009315	Magali	Rodriguez	Soto	magujose	fmagalisoto@gmail.com	6

Figura 5.7 Tabla alumnos

En la figura 5.8 se muestra la tabla de la base de datos historial

id_historial	id_alumno	id_evaluacion	id_tutorial	ultima_calificacion	ultima_evaluacion	num_intentos	calificacion
1	1	1	1	2015-02-18	2015-02-18	4	9
2	1	2	2	2015-03-10	2015-03-10	3	4
3	2	1	1	2015-02-18	2015-02-18	2	9
4	3	1	3	2015-02-19	2015-02-19	3	9
5	4	3	3	2015-02-19	2015-02-20	2	8

Figura 5.8 Tabla historial

En la figura 5.9 se muestra la tabla de la base de datos cuestionario.

id_cuestionario	id_evaluacion	id_pregunta
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	1	6
7	1	7
8	1	8
9	1	9
10	1	10
11	2	11
12	2	12
13	2	13
14	2	14
15	2	15
16	2	16
17	2	17
18	2	18
19	2	19
20	2	20

Figura 5.9 Tabla cuestionario

En la figura 5.10 se muestra la tabla de la base de datos relacion_respuestas

id_rel_res	id_pregunta	id_respuesta
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20

Figura 5.10 Tabla relacion_respuesta

En la figura 5.11 se muestra la tabla de la base de datos tipo_usuario

id_tipo_usuario	tipo_usuario	codigo_usuario
1	1	1910
2	1	5123
3	2	7863
4	2	4043
5	2	8641
6	2	6163
7	2	3245

Figura 5.11 Tabla tipo_usuario

5.3 Ejecución del plan de pruebas

En la siguiente sección, probaremos los requisitos funcionales del sistema, para ello se necesitó de la participación de usuarios en el sistema, en este caso serán los practicantes del Dr. Marco Antonio Cruz Chávez en CIICAp sus nombre son: Elizabeth Erazo Román, Areli López Orduño y Alfredo Mina Adán, que ayudaron a ejecutar este plan de pruebas. A continuación se muestra la tabla 11 correspondiente al formato del requisito funcional FN1 para el plan de pruebas.

Tabla 11. FN 1.Panel de control		
Datos generales de la prueba		
Requisito a probar: Panel de control	Ejecutor: Elizabeth Erazo Román	
Fecha y hora: 12 de Febrero del 2015,10:15am	Evaluador: José Manuel Patiño Gutiérrez	
Desarrollo		
Objetivo:	Probar el requisito funcional panel de control y su correcto funcionamiento.	
Condiciones de ejecución:	El usuario, debe estar registrado en la base de datos, para que inicie sesión como administrador.	
Entradas:	Nombre de usuario: "ciicap" Contraseña: "ciicap" Tipo de usuario: "1" código administrador: 4393	
Acciones 1. El usuario deberá ingresar al sistema. 2. Dirigirse al menú e iniciar sesión. 3. Ingresar el tipo de usuario "1" y código de administrador. 4. Ingresar su usuario, y contraseña. 5. El usuario verá el panel de control.	Resultados esperados 1. Se mostrará la página de inicio, junto con el menú. 2. Se mostrará el inicio de sesión de alumnos, el cual también permite iniciar sesión con el código y el tipo de usuario. 3. Se dirigirá al usuario a un nuevo inicio de sesión si el usuario ingresó correctamente. 4. Sí el usuario ingreso correctamente su usuario y contraseña podrá ver el panel de control.	R e s u l t a d o s obtenidos 1. El sistema mostró la página principal correctamente. 2. Se mostró correctamente el menú. 3. Se ingresó correctamente con el código y tipo de usuario. 4. Se inició sesión correctamente. 5. Se mostró el panel de control.
Evaluación de la prueba:	La prueba no presentó ninguna anomalía.	
Acciones correctivas:	Ninguna	

En la figura 5.12 se muestra la página correspondiente al Panel de control.



Figura 5.12 Panel de control.

A continuación se muestra la tabla 12 correspondiente al formato del requisito funcional FN2 gestión de tutoriales para evaluar en el plan de pruebas.

Tabla 12. FN 2. Gestión de Tutoriales	
Datos generales de la prueba	
Requisito a probar: Gestión de tutoriales	Ejecutor: Elizabeth Erazo Román
Fecha y hora: 10 de Marzo del 2015,11:30am	Evaluador: José Manuel Patiño Gutiérrez
Desarrollo	
Objetivo:	Probar el requisito funcional gestión de tutoriales y su correcto funcionamiento, para ello a través del panel de control se insertará un nuevo tutorial.
Condiciones de ejecución:	El usuario, debe estar registrado en la base de datos, para que inicie sesión como administrador.
Entradas:	Nombre de usuario: "ciicap" Contraseña: "ciicap" Tipo de usuario: "1" código administrador: 1254

Acciones	R e s u l t a d o s esperados	Resultados obtenidos
<ol style="list-style-type: none"> 1. El usuario deberá ingresar al sistema. 2. Dirigirse al menú e iniciar sesión. 3. Ingresar el tipo de usuario "1" y el código de administrador. 4. Ingresar su usuario, y contraseña. 5. Ver el panel de control. 6. Insertar o eliminar tutoriales 7. Elegir plantilla. 8. Ingresar contenido. 9. Guardar tutorial 	<ol style="list-style-type: none"> 1. Se mostrará la página de inicio, junto con el menú. 2. Se mostrará el inicio de sesión de alumnos, este también permite iniciar sesión con el código y el tipo de usuario. 3. Se dirigirá al usuario a un nuevo inicio de sesión. 4. Sí el usuario ingresó correctamente podrá ver el panel de control, con las opciones de gestión. 5. Elegir la opción de tutoriales. 6. Podrá crear un nuevo tutorial a partir de una plantilla. 7. Llenar información del tutorial. 	<ol style="list-style-type: none"> 1. El sistema mostró la página principal correctamente. 2. Se mostró correctamente el menú. 3. Se ingresó correctamente con el código y tipo de usuario. 4. Se inició sesión correctamente. 5. Se mostró el panel de control con las opciones correspondientes. 6. Se mostró la página de crear tutoriales, con las distintas opciones para el diseño de contenido. 7. Se creo un tutorial con un título, descripción obligatorios. 8. Se publicó el tutorial.
Evaluación de la prueba:	La prueba no presentó ninguna anomalía.	
Acciones correctivas:	Ninguna	

En la figura 5.13 se muestra la página correspondiente Gestión de tutoriales del panel de control.



Figura 5.13 Panel de control - Gestionar Tutoriales

En la figura 5.14 se muestra el apartado de crear tutoriales.

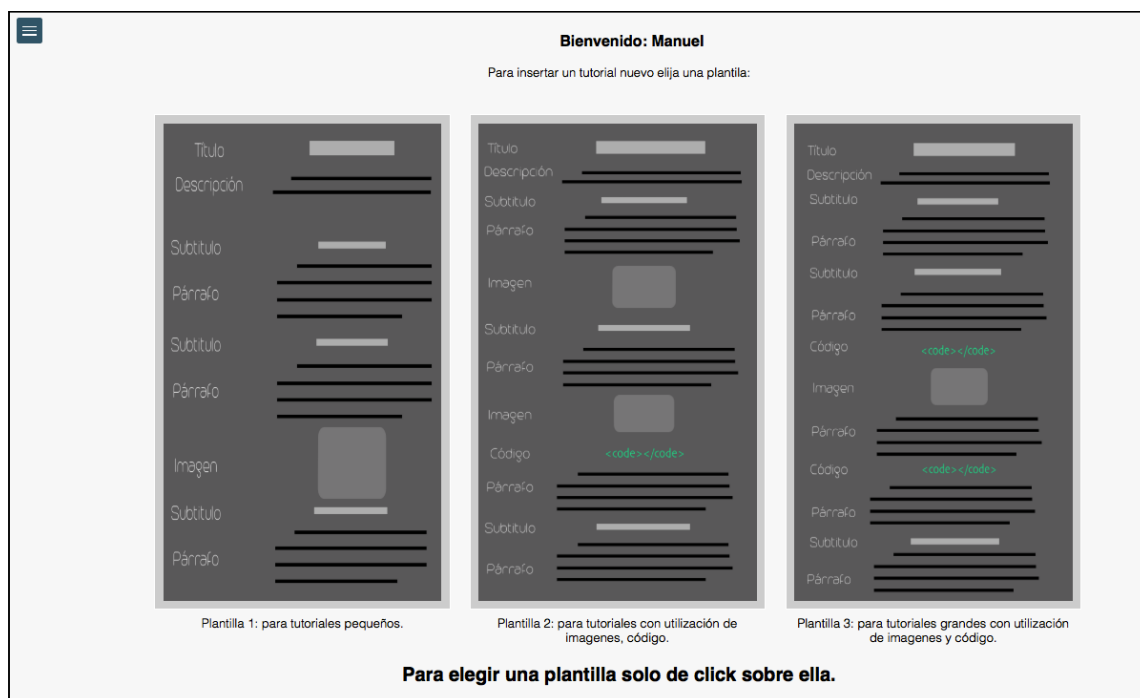


Figura 5.14 Panel de control - Gestionar Tutoriales

A continuación se muestra la tabla 13 correspondiente al formato del requisito funcional FN3 para el plan de pruebas.

Tabla 13. FN 3. Gestión de Evaluaciones		
Datos generales de la prueba		
Requisito a probar: Gestión de evaluaciones	Ejecutor: Areli Lopez Orduño	
Fecha y hora: 12 de Febrero del 2015,10:43am	Evaluador: José Manuel Patiño Gutiérrez	
Desarrollo		
Objetivo:	Probar el requisito funcional gestión de evaluaciones y su correcto funcionamiento.	
Condiciones de ejecución:	El usuario, debe estar registrado en la base de datos, para que en el sistema esté, inicie sesión como administrador.	
Entradas:	Nombre de usuario: "Usuario1" Contraseña: "QWERTY" Tipo de usuario: "2" código administrador: 1910	
Acciones 1. El usuario deberá ingresar al sistema. 2. Dirigirse al menú e iniciar sesión. 3. Ingresar el tipo de usuario "1" y código de administrador. 4. Ingresar su usuario, y contraseña. 5. Ver el panel de control. 6. Crear evaluaciones.	Resultados esperados 1. Se mostrará la página de inicio, junto con el menú. 2. Se mostrará el inicio de sesión de alumnos, en el cual se ingresará el tipo de usuario y su código. 3. Se dirigirá al usuario a un nuevo inicio de sesión, sí el usuario ingresó correctamente. 4. Sí el usuario ingresó correctamente podrá ver el panel de control, con las opciones de gestión. 5. Podrá crear una nueva evaluación. 6. Enviará la nueva evaluación.	Resultados obtenidos 1. El sistema mostró la página principal correctamente. 2. Se mostró correctamente el menú. 3. Se ingresó correctamente con el código y tipo de usuario. 4. Se inició sesión correctamente. 5. Se mostró el panel de control con las opciones correspondientes. 6. Se mostró la página de crear evaluaciones. 7. Se mostró correctamente el apartado de eliminar evaluaciones.
Evaluación de la prueba:	La prueba no presentó ninguna anomalía.	
Acciones correctivas:	Ninguna	

En la figura 5.15 se muestra un ejemplo de cómo insertar evaluaciones en el sistema.

The screenshot shows a web interface for creating an evaluation. At the top, it says "Bienvenido: Manuel". Below this, there are two main sections:

- Datos de la evaluación:** This section contains a form with the following fields:
 - Autor: Manuel
 - ¿Que es MPI? (with a sub-field for "Introducción a MPI")
 - Duración en minutos de la evaluación en horas-minutos duración máxima de 02:00horas: 01 : 20
- Preguntas de la evaluación:** This section contains a list of questions and their corresponding answers:
 - ¿Que palabras conforman las iniciales de MPI?
 - Message Passing Interface
 - Missisipi Programming Institute
 - Mensajería Parcial Interna
 - More Programming Increment
 - ¿Cuales son las características de MPI?
 - Portabilidad: multiprocesadores, multicomputar
 - Servicio, escalabilidad, presentación.
 - Rapido, multiplataforma, hilos, estructurado.
 - Multiparadigma, multilinguaje, multiprocesador

Figura 5.15 Insertar evaluaciones.

La siguiente prueba será dividida en gestión de usuarios administrador y en gestión de usuarios alumno, el requisito funcional FN4, debido a que los tipos de usuarios son distintos, para ello en la siguiente tabla 14 se muestra la evaluación del plan de pruebas para el requisito funcional FN4-Usuarios administrador.

Tabla 14. FN 4. Gestión de Usuarios - Administrador		
Datos generales de la prueba		
Requisito a probar: Gestión de Usuarios	Ejecutor: Arelí Lopez Orduño	
Fecha y hora: 12 de Febrero del 2015, 10:43am	Evaluador: José Manuel Patiño Gutiérrez	
Desarrollo		
Objetivo:	Probar el requisito funcional gestión de usuarios administradores y su correcto funcionamiento.	
Condiciones de ejecución:	El usuario, debe estar registrado en la base de datos, para que, inicie sesión como super-administrador.	
Entradas:	Nombre de usuario: "Usuario1" Contraseña: "QWERTY" Tipo de usuario: "2" código administrador: 1910	
Acciones 1. El usuario deberá ingresar al sistema. 2. Dirigirse al menú e iniciar sesión. 3. Ingresar el tipo de usuario y código de administrador. 4. Ingresar su usuario, y contraseña. 5. Ver el panel de control. 6. Elegir la opción de Administradores. 7. Elegir la opción nuevo, modificar administrador, o eliminar administrador. 8. Elegir una opción.	Resultados esperados 1. Se mostrará la página de inicio, junto con el menú. 2. Se mostrará el inicio de sesión de alumnos, el cual permite también iniciar sesión con el código y el tipo de usuario. 3. Se dirigirá al usuario a un nuevo inicio de sesión si el usuario ingreso correctamente lo anterior. 4. Si el usuario ingresó correctamente podrá ver el panel de control, con las opciones de gestión. 5. Podrá crear un nuevo usuario administrador, eliminarlo o modificarlo. 6. El sistema mostrará la opción deseada.	Resultados obtenidos 1. El sistema mostró la página principal correctamente. 2. Se mostró correctamente el menú. 3. Se ingresó correctamente con el código y tipo de usuario. 4. Se inició sesión sin ningún problema. 5. Se mostró el panel de control con las opciones correspondientes. 6. Se mostró la página de crear evaluaciones. 7. Se mostró correctamente las opciones, se eligió eliminar administrador. 8. Se mostró el apartado de eliminar administrador.
Evaluación de la prueba:	La prueba se realizó sin ningún problema.	
Acciones correctivas:	Ninguna	

En la siguiente figura 5.16 se muestra un ejemplo de eliminar un administrador.

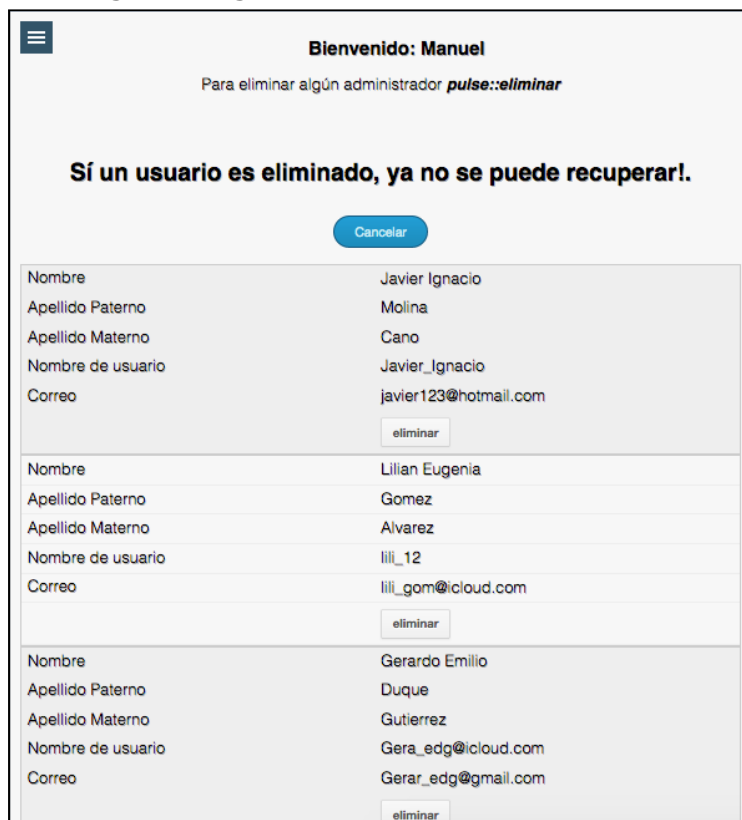


Figura 5.16 Eliminar administrador

En la siguiente tabla 15 se muestra el requisito funcional FN-4 Usuarios alumno del plan de pruebas y se comprueba su correcto funcionamiento en el sistema.

Tabla 15. FN 4. Gestión de Usuarios - Alumno	
Datos generales de la prueba	
Requisito a probar: Gestión de Usuarios	Ejecutor: Alfredo Mina Adán
Fecha y hora: 12 de Febrero del 2015,10:43am	Evaluador: José Manuel Patiño Gutiérrez
Desarrollo	
Objetivo:	Probar el requisito funcional gestión de usuarios administradores y su correcto funcionamiento.
Condiciones de ejecución:	El usuario, debe estar registrado en la base de datos, para que en el sistema esté, inicie sesión como super-administrador.

Entradas:	Nombre de usuario: "Usuario1" Contraseña: "QWERTY" Tipo de usuario: "2" código administrador: 1910	
Acciones 1. El usuario deberá ingresar al sistema. 2. Dirigirse al menú e iniciar sesión. 3. Ingresar el tipo de usuario y código de administrador. 4. Ingresar su usuario, y contraseña. 5. Ver el panel de control. 6. Elegir la opción de Alumnos. 7. Elegir una opción: nuevo, modificar, eliminar alumno, historial alumno. 8. Ver opción.	Resultados esperados 1. Se mostrará la página de inicio, junto con el menú. 2. Se mostrará el inicio de sesión de alumnos, el cual permite también iniciar sesión con el código y el tipo de usuario. 3. Se dirigirá al usuario a un nuevo inicio de sesión si el usuario ingresó correctamente lo anterior. 4. Si el usuario ingresó correctamente podrá ver el panel de control, con las opciones de gestión. 5. El sistema dirigirá al usuario a un nuevo panel, donde elegirá la opción nuevo, modificar o eliminar alumno. 6. El sistema mostrará la opción elegida.	Resultados obtenidos 1. El sistema mostró la página principal correctamente. 2. Se mostró correctamente el menú. 3. Se ingresó correctamente con el código y tipo de usuario. 4. Se inició sesión sin ningún problema. 5. Se mostró el panel de control con las opciones correspondientes. 6. Se mostró la página de alumnos. 7. Se mostró correctamente el apartado de alumnos 8. Se eligió la opción modificar alumno. 9. Se mostró la opción elegida.
Evaluación de la prueba:	La prueba se realizó sin ningún problema.	
Acciones correctivas:	Ninguna	

En la figura 5.17 se muestra modificar alumnos.

Matricula	Nombre	Apellido Paterno	Apellido Materno	Semestre
GELL150211	German Antonio	Lotero	Upegui	1
HGA150111	Hector Ivan	Gonzalez	Castano	4
BELT140612	Beatriz Elena	Osorio	Laverde	2
JUBT150312	Lorge	Uribe	Botero	1
MBLP120215	Maria Isable	Benitez	Lopez	2

[Regresar](#)

Figura 5.17 Modificar alumnos.

En la tabla 16 se muestra el requisito funcional FN 5 para el plan de pruebas

Tabla 16. FN 5. Historial del alumno		
Datos generales de la prueba		
Requisito a probar: Historial de alumno	Ejecutor: Alfredo Mina Adan	
Fecha y hora: 13 de Febrero del 2015,10:43am	Evaluador: José Manuel Patiño Gutiérrez	
Desarrollo		
Objetivo:	Probar el requisito funcional FN5 Historial de alumno y demostrar su correcto funcionamiento.	
Condiciones de ejecución:	El usuario, debe estar registrado en la base de datos, para que inicie sesión como super-administrador.	
Entradas:	Nombre de usuario: "Usuario1" Contraseña: "QWERTY" Tipo de usuario: "2" código administrador: 1910	
Acciones	Resultados esperados	Resultados obtenidos
<ol style="list-style-type: none"> 1. El usuario deberá ingresar al sistema. 2. Dirigirse al menú e iniciar sesión. 3. Ingresar el tipo de usuario y código de administrador. 4. Ingresar su usuario, y contraseña. 5. Ver el panel de control. 6. Elegir la opción de Alumnos. 7. Elegir la opción historial alumno. 8. El usuario podrá ver lista de alumnos y elegir uno. 9. El usuario podrá ver historial del alumno 	<ol style="list-style-type: none"> 1. Se mostrará la página de inicio, junto con el menú. 2. Se mostrará el inicio de sesión de alumnos, el cual permite también iniciar sesión con el código y el tipo de usuario. 3. Se dirigirá al usuario a un nuevo inicio de sesión sí el usuario ingresó correctamente lo anterior. 4. Sí el usuario ingresó correctamente podrá ver el panel de control, con las opciones de gestión. 5. El sistema dirigirá al usuario a un nuevo panel, donde elegirá la opción historial alumno. 6. Seleccionará de la lista un alumno. 7. Se mostrarán los datos de usuario alumno con su historial. 	<ol style="list-style-type: none"> 1. El sistema mostró la página principal correctamente. 2. Se mostró correctamente el menú. 3. Se ingresó correctamente con el código y tipo de usuario. 4. Se inició sesión sin ningún problema. 5. Se mostró el panel de control con las opciones correspondientes. 6. Se mostró la página de alumnos. 7. Se mostró correctamente el apartado de lista de alumnos. 8. Se seleccionó a un alumno. 9. Se mostró su historial y datos del perfil del alumno.

Evaluación de la prueba:	La prueba se realizó sin ningún problema.
Acciones correctivas:	Ninguna

En la figura 5.18 se muestra un ejemplo de mostrar el historial de alumno.

Bienvenido: Manuel
Lista de alumnos:

Matrícula	Nombre	Apellido Paterno	Apellido Materno	Semestre	
GELU150211	German Antonio	Lotero	Upegui	1	Ver
HGCA150111	Hector Ivan	Gonzales	Castaño	4	Ver
BELT140612	Beatriz Elena	Osorio	Laverde	2	Ver
JUBT150312	Jorge	Uribe	Botero	1	Ver
MBLP120215	Maria Isable	Benitez	Lopéz	2	Ver

Bienvenido: Manuel
Alumno:

Matrícula	Nombre	Apellido Paterno	Apellido Materno	Semestre	Intentos	Calificación	Título Evaluación
GELU150211	German Antonio	Lotero	Upegui	1	1	9	¿Que es MPI?
GELU150211	German Antonio	Lotero	Upegui	1	1	8	Comunicación bloqueante

[Regresar](#)

Figura 5.18 Mostrar historial de alumnos.

A continuación se muestra el requisito FN-6 en la tabla 17 para el plan de pruebas que incluye también al FN-7 Perfil de usuario. El perfil de usuario permite que una vez se inicie sesión los usuarios de tipo alumno pueden visitar su perfil.

Tabla 17. FN 6. Registro de usuarios, FN7. Perfil de usuario.		
Datos generales de la prueba		
Requisito a probar: Registro de usuarios, perfil de usuario.	Ejecutor: Elizabeth Erazo Román	
Fecha y hora: 12 de Febrero del 2015,10:43am	Evaluador: José Manuel Patiño Gutiérrez	
Desarrollo		
Objetivo:	Probar el requisito funcional FN6 y demostrar su correcto funcionamiento.	
Condiciones de ejecución:	Estar en el sistema como usuario invitado.	
Entradas:	Nombre de alumno: German, Apellido paterno: Lutero, Apellido materno: Upegui, Semestre: 1 Matricula:GELU150211, Contraseña;gelu12.	
Acciones 1. El usuario deberá ingresar al sistema. 2. Directamente desde la página de inicio elegir la opción Registrarse. 3. Llenar el formulario de registro de usuarios. 4. El usuario podrá ver su perfil.	Resultados esperados 1. Se mostrará la página de inicio, junto con el menú. 2. El sistema mostrará el formulario de registro del usuarios. 3. Al registrarse, el sistema dirigirá al usuario a su perfil. 4. El sistema mostrará el perfil de usuario. 5. Siempre que el usuario de tipo alumno inicie sesión, el sistema mostrará en el menú la opción para ver el perfil.	R e s u l t a d o s obtenidos 1. El sistema mostró la página principal correctamente. 2. Se llenó el formulario mostrado por el sistema, el cual validó las entradas. 3. Se registró al usuario. 4. Al registrarse se dirigió al usuario al perfil de usuario.
Evaluación de la prueba:	La prueba se realizó sin ningún problema.	
Acciones correctivas:	Ninguna	

A continuación en la figura 5.19 se muestra un ejemplo del perfil de usuario.

Bienvenido a tu perfil German Antonio

Datos generales:

Matrícula :GELU150211
Semestre :1

En la siguiente tabla se encuentra el historial de calificaciones:

Nombre	Título evaluación	Última evaluación	Última calificación	Número de intentos	Calificación
German Antonio	¿Que es MPI?	2015-02-18	2015-02-18	1	9
German Antonio	Comunicación bloqueante	2015-02-19	2015-02-19	1	8

Figura 5.19 Perfil de usuario de tipo alumno.

A continuación se muestra la tabla 8 con el requisito FN-8 del plan de pruebas , que incluye también el requisito FN-9 ya que para ver un tutorial se elige del Índice.

Tabla 18. FN 8. Mostrar tutoriales y FN9. Índice de tutoriales	
Datos generales de la prueba	
Requisito a probar: Mostrar tutoriales, Índice de tutoriales	Ejecutor: Elizabeth Erazo Román
Fecha y hora: 12 de Febrero del 2015,10:43am	Evaluador: José Manuel Patiño Gutiérrez
Desarrollo	
Objetivo:	Probar el correcto funcionamiento para el requisito FN 8, y FN9
Condiciones de ejecución:	Estar en el sistema como usuario invitado, o alumno.
Entradas:	En caso del usuario alumno: Matrícula: Contraseña:

Acciones	Resultados esperados	Resultados obtenidos
<ol style="list-style-type: none"> 1. El usuario deberá ingresar al sistema, en caso de ser alumno. 2. Para usuario invitado, o después de iniciar sesión, directamente desde la página de inicio elegir la opción del menú "Índice". 3. Elegir un tutorial del Índice de tutoriales. 4. El alumno podrá ver el tutorial. 	<ol style="list-style-type: none"> 1. Se mostrará la página de inicio, junto con el menú. 2. Para usuarios alumno el sistema permitirá iniciar sesión. 3. El sistema muestra el menú desde el perfil de usuario (para usuario de tipo alumno), y en la página de inicio. 4. El sistema mostrará el Índice con la lista de tutoriales. 5. Mostrar tutoriales. 	<ol style="list-style-type: none"> 1. El sistema mostró la página principal correctamente. 2. El usuario de tipo alumno inició sesión sin ningún problema. 3. El sistema mostró el Índice de tutoriales con la lista de todos los tutoriales. 4. El sistema mostró el tutorial específico, el cual fue elegido por el usuario.
Evaluación de la prueba:	La prueba se presentó sin ningún problema.	
Acciones correctivas:	Ninguna	

A continuación en la figura 5.20 se muestra un ejemplo del índice.

The screenshot shows a web application interface. At the top right, there is a red "REGISTRATE!" link. The main heading is "Bienvenido". Below it, a message reads: "Índice de temas de programación paralela con mpi. Navega por la lista ó utiliza la barra de búsqueda para encontrar un tema específico, o bien consulta el menú de opciones." There is a search bar with the placeholder text "Busca algún tema" and a blue "Buscar" button. Below the search bar is a table with three rows of tutorial entries. Each row has a "Ver" button on the right.

Título tutorial	Descripción	
Que es MPI	Se muestra la introducción de mpi, y distintas características.	Ver
Comunicación bloqueante	Comunicación bloqueante, funciones y características.	Ver
Comunicación punto a punto	Se establecen las definiciones de comunicación punto a punto	Ver

Figura 5.20 Índice

En la siguiente tabla 19 se muestra el requisito FN10 del plan de pruebas.

Tabla 19. FN 10. Buscar tutoriales o evaluaciones.		
Datos generales de la prueba		
Requisito a probar: Buscar tutoriales o evaluaciones	Ejecutor: Areli Lopez Orduño	
Fecha y hora: 12 de Febrero del 2015,10:43am	Evaluador: José Manuel Patiño Gutiérrez	
Desarrollo		
Objetivo:	Probar el correcto funcionamiento del requisito FN10	
Condiciones de ejecución:	Estar en el sistema como usuario invitado o alumno.	
Entradas:	El tutorial a buscar: En caso de ser usuario de tipo alumno: Matricula: Contraseña:	
Acciones 1. El usuario podrá ingresar al sistema. 2. En la página principal escribir en la barra de búsqueda el tutorial o evaluación a buscar. 3. En el índice y en el perfil de usuario, también se puede utilizar la barra de búsqueda. 4. El usuario podrá ver los resultados de la búsqueda.	Resultados esperados 1. El sistema permitirá al usuario iniciar sesión. 2. En sistema mostrará en la página de inició, en el índice y en el perfil de usuario la barra de búsqueda. 3. El sistema mostrará los resultados, sí es usuario invitado podrá ver las evaluaciones como resultado, pero no realizarlas.	Resultados obtenidos 1. En la página principal se utilizó la barra de búsqueda para buscar MPI: 2. Se mostró los tutoriales en una lista de acuerdo al terminó de búsqueda.
Evaluación de la prueba:	La prueba se presentó sin ningún problema.	
Acciones correctivas:	Ninguna	

En la figura 5.21 se muestra la búsqueda realizada.

Bienvenido REGISTRATE!

Para realizar alguna actividad utiliza la barra de búsqueda, o bien consulta el menú de opciones.

Busca algún tema

Buscar

Bienvenido

Índice de temas de programación paralela con mpi. Navega por la lista ó utiliza la barra de búsqueda para encontrar un tema específico, o bien consulta el menú de opciones.

Busca algún tema

Buscar

Título tutorial	Tema tutorial	
Que es MPI	Se muestra la introducción de mpi, y distintas características.	Ver
Comunicador MPI	Este tutorial versa sobre definiciones de MPI	Ver

Limpiar búsqueda

Figura 5.21 Búsqueda de MPI

El siguiente requisito del plan de pruebas es realizar evaluaciones el cual se muestra en la siguiente tabla 20.

Tabla 20. FN 11. Realizar evaluaciones		
Datos generales de la prueba		
Requisito a probar: Realizar evaluaciones	Ejecutor: Alfredo Mina Adán	
Fecha y hora: 10 de Marzo del 2015,10:43am	Evaluador: José Manuel Patiño Gutiérrez	
Desarrollo		
Objetivo:	Probar el requisito nominal FN 11 y su correcto funcionamiento.	
Condiciones de ejecución:	Estar en el sistema como usuario alumno.	
Entradas:	Matricula: GELU150211 Contraseña: gelu12	
Acciones 1. El usuario deberá ingresar al sistema. 2. En la página principal en el menú elegir la opción Evaluaciones. 3. Elegir una evaluación de la lista de evaluaciones. 4. Realizar la evaluación de acuerdo a los términos establecidos en tiempo.	Resultados esperados 1. El sistema permitirá al usuario iniciar sesión. 2. En el sistema se mostrará el menú con una opción que corresponde a evaluaciones. 3. El sistema mostrará una lista con las evaluaciones. 4. Al elegir una evaluación se mostrarán las Instrucciones de la evaluación. 5. El sistema mostrará el contenido de la evaluación, y el tiempo restante. 6. El sistema calificará la evaluación y mostrará los resultados. 7. Finalmente el sistema dirigirá al usuario a su perfil.	Resultados obtenidos 1. El sistema permitió al usuario iniciar sesión 2. Se eligió del menú la opción evaluaciones 3. Se muestra una lista con las evaluaciones disponibles. 4. Se elige una evaluación 5. Se leyeron las instrucciones y se procedió a contestar la evaluación. 6. El sistema calificó la evaluación y nos dirigió a el perfil. 7. Se mostró el historial con la nueva calificación
Evaluación de la prueba:	La prueba se presentó sin ningún problema.	
Acciones correctivas:	Ninguna	

En la figura 5.22 se muestra una evaluación.

Bienvenido

Para evaluar tus conocimientos puedes realizar alguna evaluación.
Elige alguna evaluación de las siguientes:

Título evaluación	Descripción	Duración	
¿Que es MPI?	Introducción a MPI	01:20:00	Realizar
Comunicación bloqueante	Evaluar los conocimientos sobre la comunicación bloqueante	01:00:00	Realizar

Alumno: German Antonio
Título: Comunicación bloqueante
Descripción: Evaluar los conocimientos sobre la comunicación bloqueante
Duración 01:00:00
Tiempo restante:
00:59:38

Pregunta 1
¿La comunicación bloqueante esta relacionada con?

Non-blocking

Pregunta 2
¿Si un send es bloqueante debe también?

Debe hacer un send no bloqueante

Pregunta 3
¿Un send o receive bloqueante suspende la ejecución del programa del usuario hasta que?

Tiene un impacto en los requerimientos de pr

Pregunta 4
¿Es una característica de las blocking calls?

Termina la comunicación

Pregunta 5
¿Cuántas funciones de envío (send) bloqueante son?

4

Pregunta 6

En total son 10 preguntas, con una duración total de ____.

Examine con atención cada Respuestas. Cada respuesta vale 1 (un) punto de 10 (Diez) posibles.

Cada una de las preguntas tiene 4 (cuatro) posibles respuestas. Una vez leída la pregunta seleccione la respuesta que crea correcta.

Para cada pregunta sólo hay UNA respuesta correcta. Mientras no se envíe para calificación puede cambiar la respuesta seleccionada en cualquier momento y antes del tiempo estimado.

En caso de que el tiempo termine, o envíe a calificación, las respuestas no se podrán modificar, además las respuestas que no halla contestado se enviarán como respuesta default (no implica que esta sea correcta);

Si finaliza antes del tiempo previsto, revise sus respuestas; Recuerde que la calificación mínima aprobatoria es 8.

Para Cancelar la evaluación haga click aquí, pero se tomará como un intento

Pregunta 2

Bienvenido a tu perfil German Antonio

Datos generales:

Matricula :GELU150211
Semestre :1

En la siguiente tabla se encuentra el historial de calificaciones:

Nombre	Título evaluación	Última evaluación	Última calificación	Número de intentos	Calificación
German Antonio	¿Que es MPI?	2015-02-18	2015-02-18	4	9
German Antonio	Comunicación bloqueante	2015-03-10	2015-03-10	3	4

Figura 5.22 Realizando una evaluación.

5.4 Bitácora de implantación

En este apartado se resume la bitácora de implantación del sistema, es decir, los pasos que se llevaron a cabo para dicha implantación. Para la implantación se tienen configurados dos servidores, en uno se encuentra la configuración para los servicios web, por medio del protocolo HTTP, es decir el servidor Web. El otro servidor tiene el servicio con la base de datos, ambos servidores con el sistema operativo CentOS.

1. En el Centro de Investigación en Ingeniería y Ciencias Aplicadas el día 16 de Febrero del 2015, se procedió a la implantación del sistema, en el Laboratorio de Automatización de procesos.
2. Teniendo dos computadoras con sistema operativo CentOS, se configuró la primera de ellas como servidor dedicado a servicios web mediante HTTP:
 - 2.1. Configurar J2EE, para ello se descargó KDJ y SJRE de la página oficial: www.oracle.com/technetwork/java/javase/downloads/index.html
 - 2.2. Posteriormente se descargó el IDE (Entorno de Desarrollo Integrado) de desarrollo Eclipse en su versión Kepler de la página oficial: eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/keplersr2
 - 2.3. Para el procesamiento de tecnologías de Java Server Pages (jsp) y Servlets se instaló el servidor Apache Tomcat mismo que se descargó de la página oficial: tomcat.apache.org/download-80.cgi y procedió a configurarse.
 - 2.4. Instalar Github de la página oficial: <https://github.com> y configurarlo.
 - 2.5. Por último para este primer servidor se configuró su respectiva IP proporcionada por la Universidad Autónoma del Estado de Morelos (UAEM).

3. El segundo servidor para el sistema gestor de la base de datos, se configuró lo siguiente:
 - 3.1. Una vez actualizado MySQL, se descargó phpMyAdmin de la página oficial www.phpmyadmin.net/home_page/index.php para cumplir función como SGBD (sistema gestor de base de datos) de mysql.
 - 3.2. Instalar Github de la página oficial: <https://github.com> y configurarlo.
 - 3.3. Se configuró la IP en red con el servidor Web para la conexión con el sistema.
4. El sistema se configuró mediante la cuenta de github en la cual se llevó el control de versiones del sistema. Para no afectar al sistema se hizo un “fork” (Copia del proyecto en github) al proyecto de este sistema y luego un “clon” (Descarga en el servidor de tal modo que permita hacer modificaciones) del proyecto.
5. En el servidor web se procedió a configurar en eclipse el proyecto y se configuró el build path, además de la ip de la conexión a la base de datos.
6. En el servidor de la Base de datos se realizó un fork, y un clon del sistema debido a que también la base de datos está en el proyecto de git.
7. En el servidor de la Base de datos, se importó la base de datos “BD_SIS_TUTORIALES” a phpMyAdmin y se configuró el usuario de la base de datos con permiso solo a la base de datos del proyecto.
8. Mediante Eclipse Kepler se realizó la prueba ping a la base de datos para verificar que todo se encuentre en correcto funcionamiento.
9. Se ejecutó el sistema en el servidor web.

Capítulo 6 Conclusiones y trabajo futuro

Para finalizar con el presente trabajo es necesario realizar las evaluaciones, retroalimentaciones y posibles alternativas que se tienen a futuro. En este capítulo se presentan las conclusiones del trabajo donde se resume todo el trabajo realizado, además en el apartado 6.2 se tratan los puntos a considerar para un trabajo futuro.

6.1 Conclusiones

Como conclusión se puede decir que el desarrollo de este sistema tuvo muchos beneficios, por parte del desarrollador y por parte del Centro de Investigación en Ingeniería y Ciencias Aplicadas de la UAEM. De lado del desarrollador, permitió poder llevar los conocimientos aprendidos durante la carrera y a crear nuevos conocimientos, sobre lenguajes, tecnologías, metodologías y que de alguna manera también fueron reforzados los conceptos que ya se tenían sobre estos temas.

Por parte del Centro de Investigación en Ingeniería y Ciencias Aplicadas, el desarrollo de este sistema ayuda, al mejor aprendizaje y enseñanza de MPI y de código en lenguaje C, debido a que centraliza la información y en un solo lugar es decir en el sistema, y con el panel de control se puede no tan solo crear evaluaciones que permitan demostrar los conocimientos de los usuarios, si no también crear contenido nuevo y actualizado con nuevos tutoriales con los temas de MPI para que esté siempre con la mejor información posible de tal manera que esto ahorrará mucho el tiempo de aprendizaje de esta tecnología.

Aprender acerca de MPI, era muy tedioso debido a que no se encontraba información centralizada, concisa, ni precisa, sin embargo con el sistema si estará cumpliendo con estas expectativas, debido a que el sistema contendrá los datos y con ello ser una herramienta adecuada además de hacerlo de una manera ordenada (por medio del índice y búsquedas).

Durante el desarrollo del sistema se cumplió el objetivo, aunque se tuvieron algunos problemas, como aprender el lenguaje de programación Java orientado a web, el cual fue un reto personal para poder desarrollar este sistema.

6.2 Trabajo futuro

El trabajo futuro para todo tipo de proyectos de cualquier tipo siempre, va hacer algo muy importante, ya que la filosofía de las cosas es siempre evolucionar y mejorar, además en el sentido de los proyectos de desarrollo como lo fue este sistema se necesita también mantener. Para todos estos aspectos es necesario tener muy en cuenta cuales fueron los puntos buenos y malos en el sistema, ya que de ellos se crea el área de oportunidad para mejorar.

En el aspecto evolutivo según los planes del CIICAp, este sistema podría implementarse en un futuro para ser utilizado por los profesores de CIICAP, como herramienta de enseñanza y utilizar también la aplicación de evaluaciones.

Además se pueden tomar los siguientes aspectos específicos en el sistema como áreas a mejorar en un futuro:

- Mejora de la inserción de tutoriales: Un ejemplo de la mejora en la inserción de tutoriales radica en la creación de plantillas que optimicen la carga de los tutoriales, tanto para cargar archivos provenientes de software especializado, como para aceptar mas contenido multimedia.
- Mejorar las evaluaciones: Aunque el cliente pidió como requisito que las evaluaciones fueran de la manera en la que se realizaron, podrían mejorarse para hacer las evaluaciones más dinámicas, al insertar otros tipos de preguntas.
- Mejorar el diseño de la interfaz del sistema: debido a las tendencias de diseño tan cambiantes en los últimos años, siempre es bueno tener como opción la mejora de la interfaz, debido a que mejora la experiencia de usuario y con ello permitir que el usuario se enfoque más en la información.

- Para el panel de control se recomienda el uso de gráficas que muestren el rendimiento de los tutoriales de acuerdo a los resultados de las evaluaciones.

Para el caso del mantenimiento del sistema es muy recomendable diseñar y realizar el manual técnico que apoye y documente los aspectos del sistema, detallando en él los componentes, modelos y capas del sistema. Además otro aspecto de mantenimiento sería la implementación de la función de respaldo del contenido de los tutoriales y evaluaciones automática, sin la necesidad de respaldar manualmente dicha información.

Bibliografía

Metodologías de desarrollo de software

[1] Proyectos Agiles, Beneficios de SCRUM, Universidad Autónoma del Estado de Morelos, 18 de Septiembre del 2014. <http://www.proyectosagiles.org/beneficios-de-scrum>

Técnicas de recolección de requisitos.

[2] Eco Red, Técnicas de recopilación de requisitos. Universidad Autónoma del Estado de Morelos, 29 de septiembre del 2014, [http://www.ecured.cu/index.php/T%C3%A9cnicas de recopilaci%C3%B3n de requisitos](http://www.ecured.cu/index.php/T%C3%A9cnicas%20de%20recopilaci%C3%B3n%20de%20requisitos)

[3] Pressman Roger (2010), Ingeniería del software un enfoque práctico. México DF. The Mc. Graw Hill

Arquitectura de software

[4] IEEE 1471-2000 (2011) para la arquitectura de software. IEEE

Diagramas UML Casos de uso

[5] Erich Gamma et all (2000) Design Patterns, Elements of Reusable Object-Oriented Software. México DF. ADDISON WESLEY Person Education.

Base de datos

[6] Maestros del web - Damián Pérez Valdés - Universidad Autónoma del Estado de Morelos, 31 de octubre del 2014 <http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>

[7] javier quiroz (2003) - El Modelo relacional de Bases de Datos - Universidad Autónoma del Estado de Morelos - 23 de enero del 2015 - <http://www.inegi.org.mx/inegi/contenidos/espanol/prensa/Contenidos/Articulos/tecnologia/relacional.pdf>

Model en capas

[8] CodeJobs - Cristina Elizabeth Rojas Zamora - Universidad Autónoma del Estado de Morelos, 18 de Noviembre del 2014 <http://www.codejobs.biz/es/blog/2014/01/28/la-programacion-por-capas#sthash.zvQtug4K.dpbs>

[9] Calderon Yazmín, Martínez Beatriz (2008), Desarrollo de un Sistema Administrativo de Horarios para la Facultad de Ciencias Químicas e Ingeniería, Utilizando un Modelo de Tres Capas de Cliente, Servicio Web y BD. Morelos, Mexico - Universidad Politécnica del Estado de Morelos.

Tecnologías Usadas.

[10] StarUml - MKLab 2014 - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://staruml.io>

[11] J2EE, JDK, JRE, JDBC - Oracle - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

[12] CentOS - Sync User - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <https://documentation.cpanel.net/display/1142Spanish/Escoja+un+sistema+operativo>

[13] Tomcat - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://tomcat.apache.org>

[14] Mysql - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://www.mysql.com>

[15] Mysql Workbench - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://www.mysql.com/products/workbench/>

[16] Eclipse Kepler - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://eclipse.org/kepler/>

[17] Github projet - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <https://github.com/about>

[18] W3- Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://www.w3.org/html/logo/>

[19] W3Schools - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - http://www.w3schools.com/css/css3_intro.asp

[20] W3Schools - Universidad Autónoma del Estado de Morelos - 5 de Diciembre del 2014 - <http://www.w3schools.com/js/default.asp>

[21] Libro JQuery - Universidad Autónoma del Estado de Morelos - 15 de Diciembre del 2014 - <http://librojquery.com/>

[22] UTF-8 - Universidad Autónoma del Estado de Morelos - 26 de Enero del 2015 - http://unicode.org/faq/utf_bom.html

[23] Santi Caballé y Fatos Xhafa(2007) - Aplicaciones distribuidas en Java con tecnología RMI - España - Delta publicaciones.