TEC | Tecnológico de Costa Rica

DOCUMENTO DEL DISEÑO DEL SISTEMA
IEEE STANDARD 1016

Aplicación Web del Centro Educativo Liceo El Roble

Integrantes:

- Mario Naranjo Leiva
- Jose Rodolfo Garita

LICEO EL ROBLE
HEREDIA, SANTA BARBARA
SANTO DOMINGO DEL ROBLE

08-03-17

1. Introducción

1.1 Propósito de este documento

En este documento se va a brindar especificaciones de diseño, algunos diagramas de los componentes, tales como: paquetes, actividad, clases y casos de uso. Así como diseño de la Interfaz de usuario, tecnologías elegidas para el diseño de cada componente. La especificación de este documento es referente a segmento WEB.

1.2 Alcance del proyecto

El proyecto se va a realizar para el centro educativo Liceo El Roble, donde los límites del proyecto van a ser de acuerdo a los requerimientos solicitados, la especificación de los mismos están ubicados en el documento de especificación de requerimientos, en l sección de anexos.

1.3 Definiciones, acrónimos, y abreviaciones

BD: Base de Datos.
CU: Casos de Uso.
E/R: Entidad Relación.
JS: JavaScript.
MVC: Modelo Vista y Controlador.
SDS: Software Design Specification.
SRS: Software Requirements Specification.

1.4 Referencias

Este documento se tomó del grupo de documentos brindados por el profesor Saúl Calderón, pero directamente es una plantilla de la IEEE con el código 1016.
Documento de requerimientos, este se encuentra brindado por el grupo de desarrollo de este mismo proyecto, en el mismo directorio de gitHub.

1.5 Resumen del documento

Se va a diseñar de acuerdo al patrón MVC, las clases van a ser sin atributos, donde solo se modula por los procedimientos y un mantenimiento a futuro, la única clase con atributos es la que posee la conexión a la base de datos, siendo ese el atributo, de

la misma manera todas las clases heredan de esta para sus procesamientos con la BD.

2. Descripción de la arquitectura del sistema

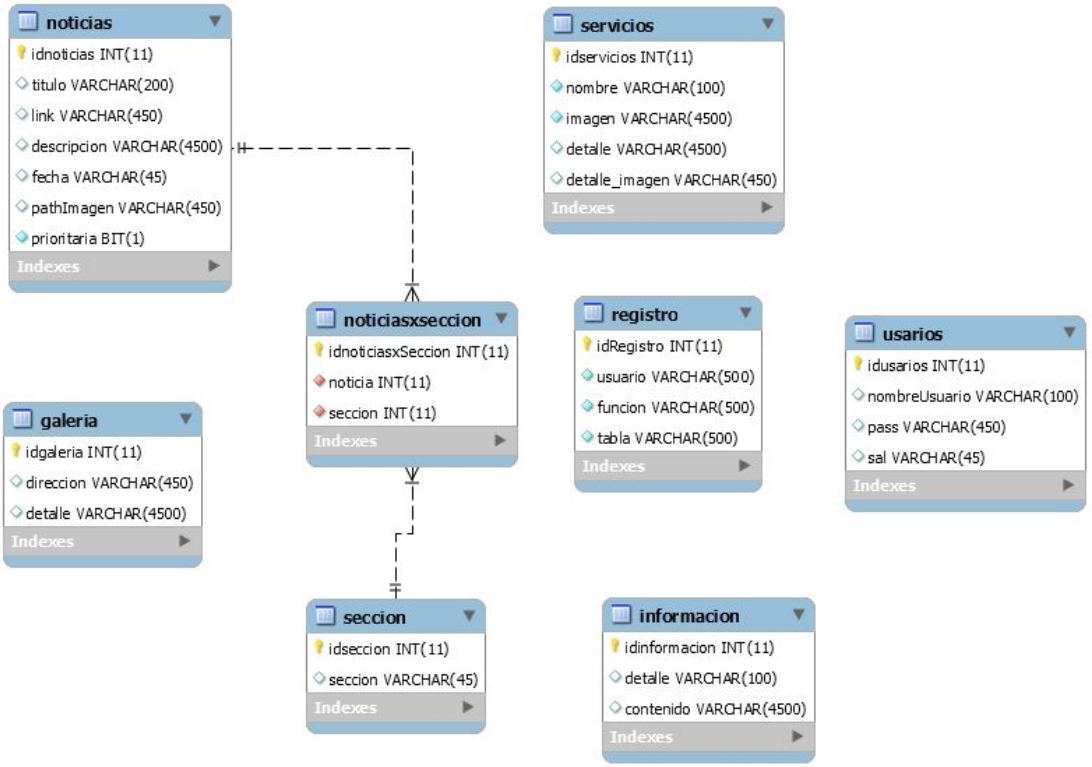
2.1 Resumen de modelos / componentes

El modelo que se va a manejar es el MVC: modelo, vista y controlador, donde las clases serán: servicio, sección, noticias, información, imagen y usuario, donde cada uno a tiene su controlador agregando como nombre Controller a cada nombre, ejemplo: ServicioController. La base de datos está modelada en un diagrama E/R, la conexión a la misma está realizada en una clase llamada DataBase.

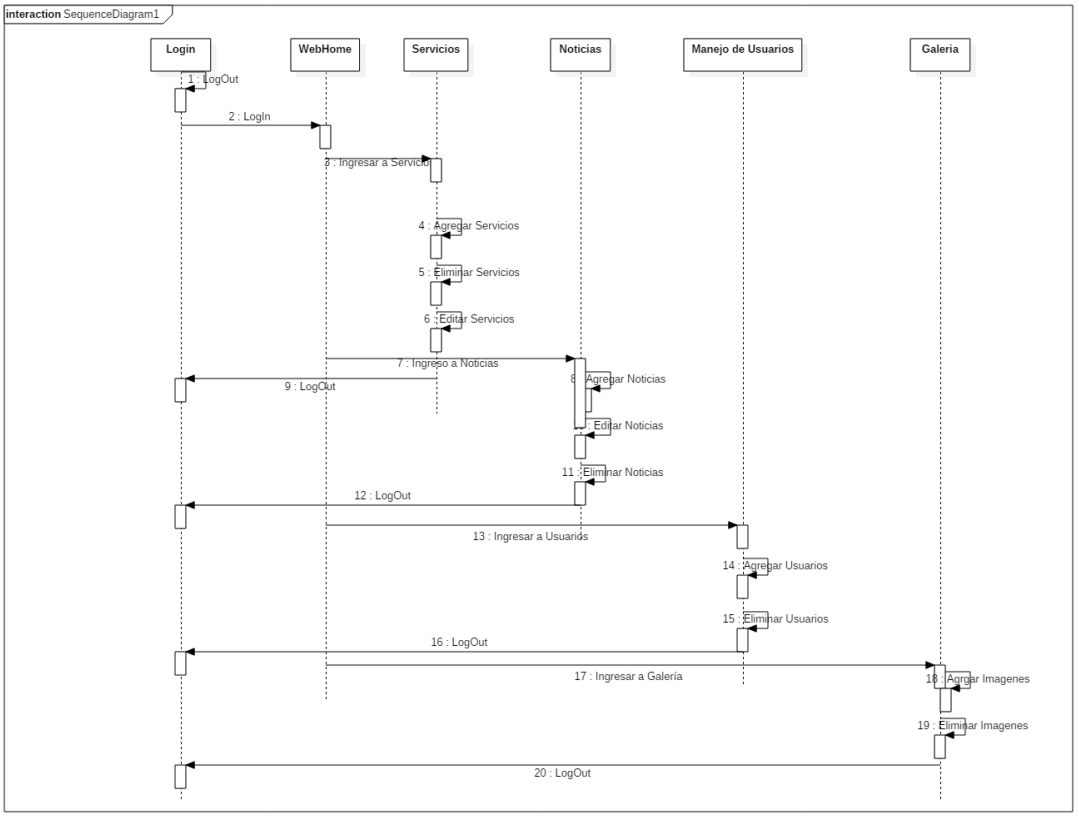
Hay dos usuarios en la aplicación: el administrador y cliente. Cada usuario tiene acceso a distintas partes de la aplicación, y además el la sección administradora hay 2 usuarios, el administrador y el estudiante, donde solo el usuario administrador - Administrador y el usuario administrador - cliente, donde el A-A tiene acceso a administrar los usuarios estudiantes administradores.

2.2 Estructuras y relaciones

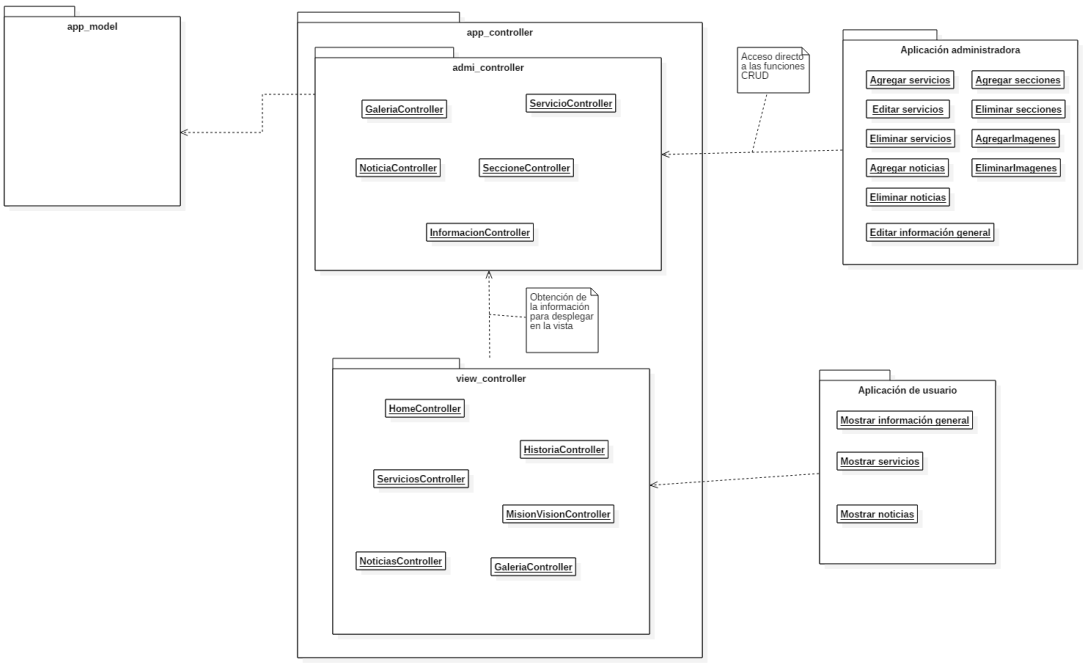
2.1. Diagrama de la Base de Datos



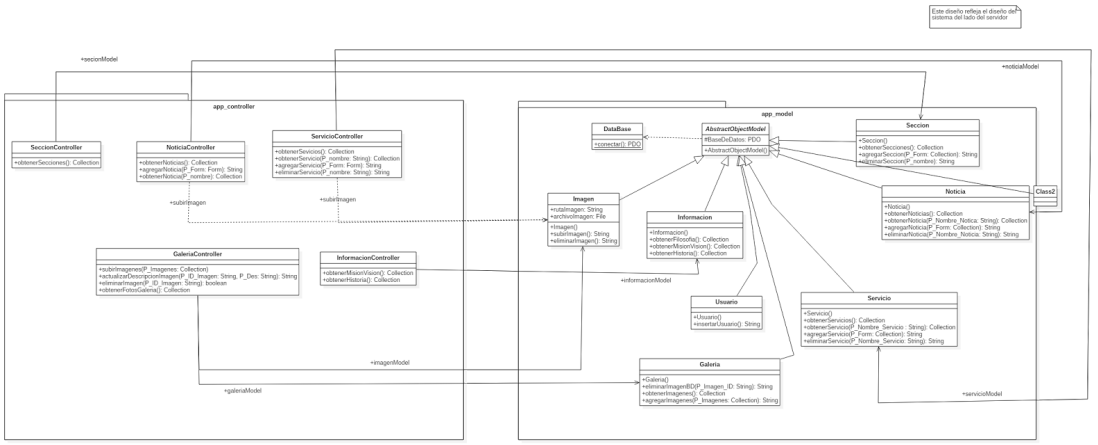
2.2. Diagrama de actividad



2.3. Diagrama de paquetes



2.4. Diagrama de clases



2.3 Cuestiones de interfaz de usuario

En la interfaz de usuario cliente, todos los que acceden al link “liceoelroble.com” van a ser tratados como usuarios cliente, y estos van a poder navegar en la página observando el contenido de la misma. Los clientes que accedan a “liceoelroble.com/MODEL/obtenerServicios.php” van a ser tratados como usuarios administradores, aquí se hace la diferencia de usuarios administrador y el estudiante, en ambos casos el login va a ser el mismo, pero el sistema reconoce el tipo de usuario, en caso de ser usuario administrador permite el acceso a Administrar usuarios, en caso de ser negativo eso lo redirecciona al HOME de administración de contenido bloqueando el acceso a la misma. El contenido editable es: eliminación, edición y agregar nuevos servicios y noticias, agregar y eliminar secciones, agregar y eliminar noticias, y editar información base: história, misión, visión y filosofía del centro.

Cada contenido verifica únicamente el minimo y maximo de caracteres, no válida el contenido.

3. Descripción detallada de los componentes

3.1 Plantilla de descripción del proyecto

La estructura de desarrollo del proyecto es así como se definió en el modelo del diseño, dónde acuerdo a los diagramas se pueden observar la estructura.

- Servicios
- Noticias

- Sección
- Usuarios
- Información
- Galería
- DataBase

Cada clase de las anteriores tiene un controlador:

- ServiciosController
- NoticiasController
- SecciónController
- UsuariosController
- InformaciónController
- GaleriaController

Y en la vista posee:

- Servicios
- Noticias
- Sección
- Usuarios
- Información
- Galeria
- Home: donde posee el direccionamiento a todas las secciones anteriores.

3.2 Modelo

El modelo contiene todas las clases base del modelo, donde posee todo el funcionamiento de cada clase, estas clases deben poseer el atributo de conexión, pero este atributo lo heredan de la clase abstracta y del modelo de base de datos.

3.3 Vista

Lleva respecto al diseño de cada clase, donde debe cumplir las operaciones de diseño de cada modelo.

3.4 Controlador

Se encarga de administrar cada segmento, es el que controla las interacciones de la vista con cada modelo, para las acciones que sean necesarias y pasar los parámetros necesarios.

4.0 Relaciones con otro proyectos

Con Google:

- Ellos usan nuestro sistema para hacer referencias en Google maps.
- Se referencia dentro del proyecto para usar los mapas del mismo.

Los font , css y JS de Google:

- ajax.googleapis.com/ajax/libs/angularjs/1.6.2/angular-route.js
- ajax.googleapis.com/ajax/libs/angularjs/1.6.2/angular-route.js
- fonts.googleapis.com/css?family=Open+Sans:400,300,600,700,800&subset=latin,latin-ext

TinyMCE:

Uso de la herramienta para edición de contenido, esta herramienta ayuda a dar formato a un texto de entrada, por ejemplo: negrita, cursiva, índices, links de imagenes….

5.0 Decisiones de diseño

En esta sección se presentan las decisiones que ayudan a entender el diseño y las tecnologías que el equipo está usando en el desarrollo del proyecto.

5.1 Tecnologías utilizadas

A continuación, se presentan las tecnología que va a ser utilizadas para la elaboración de las aplicaciones:

1. **MySQL:** Es el motor de base de datos por utilizar, se escogió por la disponibilidad de esta BD en la mayoría de servidores para el alojamiento de una aplicación web. Además, estos servidores con MySQL son más baratos y accesibles, dada la capacidad económica de la institución.
2. **MySQL Workbench**: Herramienta la diagramación de base de datos. También, permite la gestión de las bases de datos MySQL de manera local en la computadora. También se usará el gestor gráfico PhpMyAdmin, desde el servidor de alojamiento.
3. **PHP:** Lenguaje de programación del lado del servidor. Junto con MySQL, este lenguaje es mucho más común en los servidores. Además, permite la programación de servicios web, con encapsulamiento de datos en notación JSON, para la conexión a la base de datos.
4. **HTML5 y CSS3**: Lenguajes para la creación de las interfaces de usuario para la aplicación web.
5. **Responsee:** Es un framework para CSS3. Este permite crear páginas responsivas con el menor uso de líneas de código. Además, las plantillas usadas para la aplicación está inicialmente implementadas con este framework.
6. **AngularJS (Javascript):** Framework de javascript de código abierto. Se escogió para mantener la aplicación web basada en la arquitectura

de MVC (Modelo, Vista y Controlador). Este permite el consumo de servicios web, que envían los datos encapsulados en JSON. Además, su uso se recomienda, pues la curva de aprendizaje con este framework es mucho menor.

7.0 Anexos

SDS componentes

Identificación	Modelo
Tipo	Módulo
Proposito	Modularizar el código, hacer que el mismo sea más sencillo de modificar y de entender, tratando que los segmentos del modelo sea las clases finales y al agregar características solo se hagan desde aquí.
Función	Posee los modelos, las clases a utilizar en el proyecto.
Subordinado	Posee una herencia a un modelo generalizado, este modelo se llama “AbstractModel” que posee la conexión a la base de datos.
Dependencias	Cada uno de estos modelos debe ser dependiente. Quiere decir que al modificar uno no se vean afectados ni beneficiados ningún otro modelo.
Interfaces	La conexión a la base de datos va poseerla la clase general, está clase va a ser la encargada de verificar la conexión a la base de datos. El manejo de errores los maneja el html, pasando parámetros ya verificados y no en blanco (los que sean obligatorios).
Recursos	NA.
Procesamiento	De entrada del controlador manipula las clases.
Datos	Parametros.

Identificación	Vista
Tipo	Módulo
Proposito	Modularizar el código, hacer que el mismo sea más sencillo de modificar y de entender, tratando que los segmentos del modelo sea las vistas finales y al agregar características solo se hagan

	desde aquí.
Función	Posee las vistas, las que muestran el modelo.
Subordinado	NA
Dependencias	Cada uno de estos modelos debe ser dependiente. Quiere decir que al modificar uno no se vean afectados ni beneficiados ningún otro modelo. Solo se comunican entre ellos para mostrar los datos al usuarios.
Interfaces	El manejo de errores los maneja aquí (html), y mostrando los mensajes.
Recursos	NA.
Procesamiento	Muestra y envía datos. No procesa nada. Más que los tipos de entrada de texto.
Datos	Parametros.

Identificación	Controlador
Tipo	Módulo
Proposito	Administra el modelo, hacer que el mismo sea más sencillo de modificar y de entender, tratando que los segmentos del modelo sea las clases finales fáciles de administrarse y al agregar controladores solo se hagan desde aquí.
Función	Posee los controladores, la administración de las clases en el proyecto.
Subordinado	NA
Dependencias	Cada uno de estos modelos debe ser dependiente. Quiere decir que al modificar uno no se vean afectados ni beneficiados ningún otro modelo.
Interfaces	El manejo de errores los maneja el html, pasando parámetros ya verificados y no en blanco (los que sean obligatorios). manipula la entrada de los datos, haciendo que las clases del modelo solo creen el objeto correspondiente y haciendo la verificación de los datos de entrada que el html no pueda verificar.
Recursos	NA.
Procesamiento	De entrada la vista posee los datos.
Datos	Parametros.