

UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA

UNAN- LEÓN

FACULTAD DE CIENCIAS Y TECNOLOGÍA

DEPARTAMENTO DE COMPUTACIÓN



**TESIS PARA OPTAR AL TÍTULO DE INGENIERÍA EN SISTEMAS DE
INFORMACIÓN**

Tema:

Desarrollo de una aplicación web para la gestión de registros académicos del colegio Calasanz – León, utilizando el framework Laravel de PHP, en el periodo Enero 2015 - Julio 2016.

Autores:

- Yajaira Martina Solís Mayorga.
- Erika Julia Rivas Duarte.
- Jader Efraín Sevilla Centeno

Tutor: MSc. Miguel Ángel Bárcenas Lezama.

León, Noviembre 2016.

“A La Libertad Por La Universidad”

Dedicatoria

A mis padres Cándida Rosa Ramírez y Reynaldo Carmelo Solís por haberme apoyado en todo momento, por sus valores por la motivación constante que me encaminaron a ser una persona de bien, pero más que nada por su amor.

A mis hermanos Isabel, José y Alexander por su ayuda y comprensión por estar conmigo y apoyarme siempre.

Yajaira Martina Solís Mayorga

Dedicatoria

A mi hija Jaheri Jubana Vargas Rivas por ser la fuerza que me empuja a seguir luchando por alcanzar mis sueños.

A mi madre Juana Venicia Duarte Herrera y a mi padre Evaristo Rivas Sánchez por ser ejemplo de amor y lucha para sus hijos.

A mis hermanas Yamileth y Gladis, a mi hermano Andrés porque los cuatro hemos luchado por alcanzar los sueños que cuando niños nos propusimos.

A mi compañero de vida Jasson Vargas porque asumimos esta lucha junto.

Erika Julia Rivas Duarte

Dedicatoria

Principalmente a Jehová Dios, por ser mi amigo fiel y el ser que me ha brindado las fuerzas, quien ha guiado mis pasos y me ha permitido llegar hasta donde estoy.

A mis padres y familiares, por su ayuda incondicional en todo momento y por sus esfuerzos amorosos, permitiendo de esa manera que yo pudiera alcanzar otra meta más en mi vida.

A mis maestros, por su gran paciencia y dedicación y porque son quienes siempre nos impulsan y motivan a dar lo mejor de nosotros mismos y cumplir nuestros sueños.

A mis amigos, por ser quienes siempre estaban a mi lado, aunque las cosas nos siempre marcharan bien, y por ser nuestros compañeros en el mismo derrotero.

Jader Efraín Sevilla Centeno

Agradecimientos

A Dios por su infinita misericordia y amor. Por haberme permitido llegar hasta este punto y por haberme dado salud y guiar mis pasos.

A mi familia que han sido mi soporte y compañía durante todo el periodo de estudio.

A mis compañeros Erika Rivas y Jader Sevilla por el apoyo mutuo en nuestra formación profesional.

A nuestro tutor MSc. Miguel Bárcenas Lezama que siempre estuvo dispuesto a ayudarnos en todo momento hasta finalizar nuestra monografía.

A Luis David Solórzano Paredes por haberme brindado su apoyo al realizar este trabajo.

Yajaira Martina Solís Mayorga

Agradecimientos

A Dios porque tu misericordia ha sido infinita conmigo y porque me ha permitido alcanzar un éxito más.

A mi padre y a mi madre por haber luchado para que yo pudiera ser una profesional.

A mi compañero de vida porque me ha apoyado para que continúe hacia adelante y por haber perseverado a mi lado a pesar del tiempo y la distancia.

A mis compañeros de grupo Yajaira y Jader porque siempre que pedí su ayuda sabían decirme que sí y por el recorrido de todos estos años que juntos hemos realizado.

A nuestro tutor MSc. Miguel Bárcenas Lezama, por la paciencia que siempre tuvo con nosotros y el tiempo que nos dedicó durante el transcurso de esta carrera.

Erika Julia Rivas Duarte

Agradecimientos

A Jehová Dios, por sus infinitas bendiciones en todo momento, por brindarme salud, fortaleza y valor para poder alcanzar mis objetivos y metas, además de su infinita misericordia.

A mis padres Tomas Sevilla y Bernarda Centeno, gracias por el amor y el apoyo incondicional, sincero y valioso de todos los días. Sin ustedes no hubiese podido culminar esta grandiosa etapa.

A nuestros maestros, en especial a nuestro tutor MSc. Miguel Ángel Bárcenas Lezama, por ser parte fundamental de este crecimiento como personas y como estudiantes. ¡Gracias por compartir su tiempo y conocimiento!

A todos mis amigos y compañeros, que han estado conmigo a lo largo de este tiempo. Sin su alegría, tiempo y consejos diarios no podría haber llegado hasta hoy.

A mi amigo Luis Solórzano quien siempre me ha apoyado y más en nuestro trabajo monográfico que culmina con nuestra carrera. Fuiste de mucha ayuda en todo. ¡Gracias!!!

Jader Efraín Sevilla Centeno

Resumen

El colegio Calasanz es una institución educativa apegada a la Orden de los clérigos regulares pobres de la Madre de Dios de las Escuelas Pías (Ordo Clericorum Regularium pauperum Matris Dei Scholarum Piarum, Sch. P. o S. P.), más conocidos como escolapios, llamados también piaristas en países germanoeslavos y anglosajones y calasancios en algunos iberoamericanos.

Es una orden religiosa fundada por San José de Calasanz (1557-1648) en el siglo XVII, para dar respuesta a la necesidad educativa de los niños pobres de Roma, desde donde se extendió por toda Italia y Europa Central. Más tarde hacia américa y otras regiones del mundo.

El colegio Calasanz al igual que muchas instituciones trabaja con una gran cantidad de información proveniente de los diferentes procesos académicos que se realizan, esto genera un problema, ya que la única forma de gestionar correctamente esta información es digitalizarla. Actualmente cuenta con un sistema de registro, que está presentando algunas deficiencias a la hora de manejar toda la información existente.

Por esta razón es necesario desarrollar una aplicación para agilizar los procesos de registro académicos. Esto se logrará ya que la aplicación será manipulada por un administrador a la hora de realizar matriculas, registrar notas, registrar a los docentes, registrar materias, ver notas, imprimir certificados de notas, editar una nota y editar su contraseña cuando lo considere necesario, reduciendo así el tiempo de procesamiento de la información. El objetivo del desarrollo de esta aplicación consiste en facilitar el manejo de la información de registro académico del colegio, agilizando los procesos de manipulación de dicha información.

Índice

Contenido

1. Introducción.....	16
2. Planteamiento del problema.....	17
3. Antecedentes	18
4. Justificación.....	20
5. Objetivos	22
5.1. Objetivo General	22
5.2. Objetivos específicos	22
6. MARCO TEÓRICO.....	24
6.1 Educación	24
6.2 Descripción de las herramientas que se utilizarán	25
6.2.1 Descripción del servidor XAMPP	25
6.2.2 Descripción de la herramienta MYSQL.....	25
6.2.3 Descripción de phpMyAdmin	26
6.2.4 Descripción del framework Laravel de PHP.....	27
6.2.5 Descripción del framework CSS Gumby.....	28
6.2.6 Descripción de MD5.....	28
6.2.7 Descripción del editor de texto sublime Text	28
6.2.8 Descripción de PHP	29
6.2.9 Descripción de Ajax	30
6.2.10 JQuery	31
7. DISEÑO METODOLOGICO	32
7.1 Ciclo de vida.....	34
7.1.1 Etapa de inicialización	34
7.1.2 Etapa de iteración	34
8. Especificación de requisitos	35
8.1 Introducción	35
8.2 Propósito.....	36
8.3 Ámbito del Sistema	36

8.4 Definiciones y Acrónimos.....	37
8.4.1 Definiciones	37
8.4.2 Acrónimos.....	38
8.5 Referencias.....	39
8.6 Visión General del Documento.....	39
9. Descripción General.....	39
9.1 Perspectivas del producto.....	39
9.2 Funciones del producto.....	40
9.3 Características de los Usuarios.....	40
9.4 Restricciones.....	41
10. Requisitos específicos.....	41
10.1 Requisitos comunes de las interfaces.....	42
10.1.1 Interfaces de usuario	42
10.1.2 Interfaces Hardware.....	42
10.1.3 Interfaces Software	42
10.1.4 Interfaces del Sistema	42
10.1.5 Interfaces de comunicación	42
10.2 Requisitos funcionales	43
10.2.1 Requisito funcional 1.....	43
10.2.2 Requisito funcional 2.....	44
10.2.3 Requisito funcional 3.....	45
10.2.4 Requisito funcional 4.....	46
10.2.5 Requisito funcional 5.....	47
10.2.6 Requisito funcional 6.....	48
10.2.7 Requisito funcional 7.....	49
10.2.8 Requisito funcional 8.....	50
10.2.9 Requisito funcional 9.....	51
10.2.10 Requisito funcional 10.....	52
10.2.11 Requisito funcional 11.....	53
10.2.12 Requisito funcional 12.....	54
10.2.13 Requisito funcional 13.....	55
10.2.14 Requisito funcional 14.....	56

10.2.15 Requisito funcional 15.....	57
10.3 Requisitos de Rendimiento	58
10.3.1 Recursos disponibles.....	58
10.4 Atributos de la aplicación	58
10.4.1 Seguridad	58
11. Diagramas de secuencia	59
11.1 Diagrama de Secuencia 1: Inicio de sesión de un usuario.....	59
11.2. Diagrama de secuencia 2: Registrar notas.	60
11.3 Diagrama de secuencia 3: Registrar matricula.....	61
11.4 Diagrama de secuencia 4: Registrar profesores.	62
11.5 Diagrama de secuencia 5: Registrar materia.	63
12. ANEXOS	65
12.1 Modelo Entidad – Relación	65
12.2 Diseño de la interfaz	66
12.2.1 Pantalla principal de la aplicación web.	66
12.2.2 Ingreso del Usuario y Contraseña.....	67
12.2.3 Edición de contraseña.	69
12.2.4 Listado de alumnos existentes.....	70
12.2.5 Pestaña de “Registrar”	71
12.2.6 Pestaña de “Notas”	76
13. CONCLUSIONES.....	80
14. BIBLIOGRAFIA	81
Referencias	81

1. Introducción

El presente trabajo está dirigido a mitigar problemas con el manejo de la información de registros académicos del colegio Calasanz, el cual es una institución educativa, católica y privada ubicada en la ciudad de León - Nicaragua. Esta institución cuenta con una gran cantidad de estudiantes, que deben ser atendidos en el área de registro académico.

Por lo tanto, esta área genera gran cantidad de datos los cuales deben ser almacenados. Una parte de estos son digitalizados, pero hay grupos que su control se lleva manualmente.

Para ello la institución cuenta con un sistema de registro académico estándar que permite el control de los grupos existentes desde el tiempo que se creó el sistema.

2. Planteamiento del problema

El problema de toda institución moderna se basa en la eficiencia de procesar su información, este problema lo tiene el colegio Calasanz en el área de registro académico, la información asociada a los nuevos grupos de clase que se abrieron después de creado el sistema que utilizan, no está siendo controlada, ni manejada digitalmente.

Otro problema es que al realizar una tarea 100% manual además de ser tedioso, ineficiente y muy tardado, no proporciona integridad en los datos obtenidos, es decir, no se sabe si los datos son correctos. Relacionado con este problema está el hecho que todo proceso está sujeto a errores humanos perjudicando así la calidad respecto a la atención que se brinda a los estudiantes y padres de familia. Hasta la fecha de realización del presente trabajo no existe un sistema completo que resguarde toda la información que maneja el área de registro académico.

Para contribuir con la solución de este problema se pretende desarrollar una aplicación de registro académico en donde se pueda almacenar y llevar el control total de la información, para mejorar los servicios que ofrece el colegio. Además, la aplicación deberá validar la entrada del administrador, verificando que el usuario y contraseña sea correctos, esto proporciona la seguridad de que solo la persona indicada podrá manipular la información que este registrada en dicha aplicación.

Esta aplicación se ejecutará en cualquier equipo con un navegador web, y se desarrolla con el framework Laravel de PHP. La aplicación mostrara formularios obteniendo la información de los diferentes procesos que se realizan en el área de registro académico, apoyando de esta manera a mejorar la calidad de los servicios ofrecidos.

3. Antecedentes

En muchos países del mundo se han desarrollado sistemas de registro académico, tales como la universidad de los Andes en donde se han venido realizando sistemas de información en los que se gestionan actividades académicas, de investigación, tanto de docentes como de estudiantes, en el departamento de computación de la universidad de los Andes en la Escuela de Ingeniería en Sistema de Información para el control académico, de Investigación y Extensión del Departamento de Computación de EISULA (SIDEKOM). (Henry A. Rodriguez C., 2008)

En Nicaragua se han implementado varios sistemas de registro académico, como es el caso de la UNAN-LEON. En el año 2002 se contaba con un sistema de registro académico llamado REGACAD, dicho sistema permitía almacenar y visualizar notas de los estudiantes; para el año 2006 se crea el nuevo sistema y actualmente funcional llamado SIGACAD¹, éste entra en función paralelo al antiguo sistema; en el año 2008 se integran nuevos módulos como Planificación Académica, y Registro Académico, lo cual deja obsoleto el antiguo sistema y queda solo funcional a nivel de consulta (este contiene los registros de estudiantes matriculados durante el funcionamiento de REGACAD), en base a esto se pueden generar las actas para el estudiante.

El colegio Calasanz, desde sus inicios ha llevado fielmente el registro de las calificaciones de los estudiantes de manera tradicional. La era digital ha propuesto un nuevo camino al cual nos vemos necesitados a recorrer por la fluidez de la información tanto para uso docente como del conocimiento de los actores que intervienen en la formación integral del estudiante.

¹SIGACAD, es el Sistema de Gestión Académica que utiliza la UNAN-León. El objetivo principal del sistema es proveer a la universidad de mecanismos automatizados que faciliten la planificación, organización, gestión y control académico. (UNAN-León. (2006). <https://www.sigacad.unanleon.edu.ni/> ; UNAN-León. (2006). <https://www.sigacad.unanleon.edu.ni/> .)

En años anteriores en el colegio se manejaba la información manualmente o en hojas de Excel², por ejemplo. Luego se creó un sistema de registro, el cual fue funcional y eficiente, en su momento le permitió al colegio mantener segura y estable su información.

Posteriormente el colegio empezó a crecer y aumentar el número de estudiante, entonces fue necesario abrir nuevos grupos de clases, pero el sistema estándar ya no registraba a estos porque no tenía la opción de extender el número de grupos, entre otros inconvenientes que empezó a presentar.

Y para cumplir con las exigencias de la administración académica del colegio hemos propuesto crear esta nueva aplicación de registro académico que venga a facilitar y agilizar el proceso académico del colegio.

²Excel es un programa informático desarrollado y distribuido por Microsoft Corp. Se trata de un software que permite realizar tareas contables y financieras gracias a sus funciones, desarrolladas específicamente para ayudar a crear y trabajar con hojas de cálculo.
(<http://definicion.de/excel/>, 2015)

4. Justificación

El colegio Calasanz de la ciudad de León como centro de educación privado maneja una cantidad masiva de información dependiente del gran número de estudiantes que presenta actualmente.

Este centro cuenta con un sistema de registro académico, al cual no se la ha podido dar mantenimiento, dejando fuera mucha información relevante del mismo.

Debido a la problemática que genera el uso del sistema actual como es la lentitud del proceso, la falta de seguridad en los datos y el descontrol, nace la idea de implementar una aplicación informática que controle el proceso de gestión académica del colegio y así la posibilidad para mitigar la problemática actual.

Por tal motivo, el presente proyecto centra su objetivo en crear una alternativa que agilice los procesos de manejo de información en el área de registro académico, por ende se reducirá el tiempo de manipulación de la información y disminuirá la carga laboral del personal administrativo de la institución.

La aplicación se va a desarrollar con los frameworks³ LARAVEL⁴ de php⁵, CSS gumby, haciendo uso del servidor web XAMPP⁶, con el editor de texto

³Estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. (www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf, 2015)

⁴ Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". (<http://laravel.com/>, 2015)

⁵ PHP es un lenguaje de programación de propósito general popular que es especialmente adecuado para el desarrollo web. (<http://www.w3.org/html/>, 2014)

⁶ XAMPP es una distribución de Apache completamente gratuita y fácil de instalar que contiene MySQL, PHP y Perl. (<https://acrobat.adobe.com/la/es/products/about-adobe-pdf.html>, 2015)

SUBLIME TEXT⁷, utilizando los lenguajes de programación HTML⁸, PHP y usando el sistema de encriptado de contraseñas MD5⁹.

Este proyecto también beneficiará de manera indirecta a los estudiantes, padres de familia y al personal mismo que labora en ésta institución con la mejora de la calidad de los servicios mencionados anteriormente.

⁷ Es un editor de texto sofisticado para el código, marcado y prosa. **Fuente especificada no válida.**

⁸HTML es el lenguaje básico de la Web para crear documentos y aplicaciones para que todos puedan usar en cualquier lugar. (<http://www.w3.org/html/>, 2014)

⁹ Message-Digest 5, conocido simplemente como MD5, es una de las maneras más rápidas y sencillas de añadir seguridad a los archivos y mensajes que envía y transferencia. **Fuente especificada no válida..**

5. Objetivos

5.1. Objetivo General

- Desarrollar una aplicación en el entorno web para la gestión de registros académicos del colegio Calasanz – León, utilizando el framework Laravel de PHP, en el periodo Enero 2015 – Julio 2016.

5.2. Objetivos específicos

- Diseñar una base de datos para una aplicación de registro académico utilizando sistemas de base de datos de libre distribución.
- Desarrollar interfaces gráficas para la gestión de datos académicos.
- Generar reportes de notas.

Resultados esperados

En el siguiente apartado se describen los resultados que se deben obtener una vez finalizado todas las etapas del proyecto.

- Diseño de las interfaces de la aplicación, las cuales permitirán especificar como va a interactuar la aplicación con el usuario y con el servidor.
- Diseño de una aplicación de registro académico con el framework Laravel de PHP con funciones que permita comunicar la aplicación con el servidor de base de datos.
- Realizada la codificación del sistema, este se aplicará de acuerdo a las interfaces diseñadas, permitirá interactuar con el usuario para la introducción de datos y con el servidor para validad y guardar estos datos.
- Realizadas pruebas del sistema, se realizará un informe donde se especificarán los errores de código y diseño que los usuarios han encontrado. Así como recomendaciones o mejoras.

6. MARCO TEÓRICO

6.1 Educación

En el siglo XXI la permanencia se encuentra en el cambio y en la transformación. El cambio es la regla en la “Era de la información¹⁰”. Aquello que no evoluciona produce inestabilidad, crisis y está predestinado a desaparecer. Este nuevo entorno crea necesidades específicas en la sociedad que la educación debe cubrir para adaptarse a un medio en el que la única constante es el cambio.

No debemos olvidar que la educación busca dentro sus objetivos últimos la formación integral del ser humano, entendido como un ser de necesidades, habilidades y potenciales, busca intervenir en las Dimensiones Cognitivas (conocimientos) Axiológica (valores) y Motora (habilidades y destreza), para mejorar la calidad de vida.

En Nicaragua la integración de la tecnología con la educación es aun débil, dado al poco desarrollo económico del país, factor que afecta en todos los ámbitos sociales, limitando la incorporación de estas instituciones con la nueva tecnología. El colegio Calasanz se encuentra ubicado en el barrio Sutiava en la ciudad de León, tiene como responsabilidad formar con excelencia académica y cristiana a futuros profesionales. Este colegio imparte únicamente el turno matutino a niveles de preescolar, primaria y secundaria.

¹⁰ Nombre que recibe el período aproximado que, sucede a la era espacial y antecede a la economía del conocimiento y va ligada a las tecnologías de la información y la comunicación. (<http://es.slideshare.net/laemoyelemo/era-de-la-informacion-5184737>, 2015)

6.2 Descripción de las herramientas que se utilizarán

6.2.1 Descripción del servidor XAMPP

XAMPP es el entorno más popular de desarrollo con PHP, XAMPP es una distribución de Apache completamente gratuita y fácil de instalar que contiene MySQL, PHP y Perl¹¹. El paquete de instalación de XAMPP ha sido diseñado para ser increíblemente fácil de instalar y usar.

6.2.2 Descripción de la herramienta MYSQL

MySQL es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de tablas que contienen datos. Para agregar, acceder y procesar datos guardados en un computador se necesita un administrador como MySQL server. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de base de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un solo archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

MySQL es software de fuente abierta lo cual significa que es posible para cualquier persona bajar el código fuente de MySQL y usarlo sin pagar así como estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir qué puede hacer y que no puede hacer con el software en diferentes situaciones, pero si requiere introducir código MySQL en aplicaciones comerciales, se debe comprar una versión comercial licenciada.

¹¹ Es un lenguaje de programación. Perl toma características del lenguaje C, del lenguaje interpretado bourne shell (sh), AWK, sed, Lisp y, en un grado inferior, de muchos otros lenguajes de programación. **Fuente especificada no válida.**

6.2.3 Descripción de phpMyAdmin

PhpMyAdmin es una herramienta de software libre escrito en PHP, la intención de manejar la administración de MySQL a través de Internet. PhpMyAdmin es compatible con una amplia gama de operaciones en MySQL. Utilizado con frecuencia en operaciones (gestión de bases de datos, tablas, columnas, relaciones, índices, usuarios, permisos, etc.) se puede realizar a través de la interfaz de usuario, mientras que usted todavía tiene la capacidad de ejecutar directamente cualquier sentencia SQL.

6.2.3.1 Características

- Interfaz web intuitiva.
- Apoyo a la mayoría de las características de MySQL: buscar y borrar bases de datos, tablas, vistas, campos e índices, crear, copiar, bajar, cambiar el nombre y modificar bases de datos, tablas, campos e índices.
- Servidor de mantenimiento, bases de datos y tablas, con propuestas sobre la configuración del servidor.
- Administrar usuarios y privilegios de MySQL.
- Gestionar los procedimientos almacenados y disparadores.
- Importar datos de CSV y SQL¹².
- Exportar datos a varios formatos: CSV, SQL, XML¹³, PDF¹⁴, ISO / IEC 26300¹⁵, Hojas de cálculo y otros.
- La administración de varios servidores.

¹²SQL corresponde a la expresión inglesa Structured Query Language (entendida en español como Lenguaje de Consulta Estructurado), la cual identifica a un tipo de lenguaje vinculado con la gestión de bases de datos de carácter relacional que permite la especificación de distintas clases de operaciones entre éstas. (<http://definicion.de/sql/>, 2015)

¹³ XML, siglas en inglés de eXtensible Markup Language ('lenguaje de marcas extensible'), es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible. **Fuente especificada no válida.**

¹⁴ PDF es un formato de archivo utilizado para presentar e intercambiar documentos de forma fiable, independiente del software, el hardware o el sistema operativo. (<https://acrobat.adobe.com/la/es/products/about-adobe-pdf.html>, 2015)

¹⁵ Formato de Documento Abierto para Aplicaciones Office. **Fuente especificada no válida.**

6.2.4 Descripción del framework Laravel de PHP

Laravel es un nuevo y poderoso **Framework PHP** desarrollado por **Taylor Otwell**, que promete llevar el lenguaje PHP a un nuevo nivel. Desarrollar aplicaciones usando Laravel es muy sencillo, fundamentalmente debido a su expresiva sintaxis, sus generadores de código, y su ORM incluido el paquete llamado Eloquent ORM¹⁶.

Laravel, propone una forma de desarrollar aplicaciones web de un modo mucho más ágil. Por ejemplo, en Laravel opcionalmente podemos usar el **patrón de diseño MVC**¹⁷ (Modelo-Vista-Controlador) tradicional.

Trabajando con laravel como framework de base en el desarrollo de los proyectos web freenlance obtenemos:

- Reduccion de costos y tiempos en el desarrollo y mantenimiento.
- Curva de aprendizaje relativamente baja (en comparación con otros framework php).
- Es modular y con un amplio sistema de paquetes y drivers con el que se puede extender la funcionalidad de forma fácil, robusta y segura.
- Hace que el manejo de los datos en Laravel no sea complejo; mediante Eloquent (ORM basado en el patrón active record) la interacción con las bases de datos es totalmente orientada a objetos, siendo compatible con la gran mayoría de las bases de datos del mercado actual y facilitando la migración de nuestros datos de una forma fácil y segura.
- Permite la creación de consultas robustas y complejas.

¹⁶El elocuente ORM incluye con laravel ofrece una hermosa aplicación, simple ActiveRecord para trabajar con su base de datos. Cada tabla de base de datos tiene un "modelo" correspondiente que se utiliza para interactuar con esa tabla.**Fuente especificada no válida.**

¹⁷ El modelo–vista–controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones.**Fuente especificada no válida.**

6.2.5 Descripción del framework CSS Gumby

Gumby, un framework para diseñar páginas web con responsive¹⁸. Está basado en Grid's¹⁹ que te permite personalizar y moldear el diseño de una página adaptándose a tus necesidades, como por ejemplo al tamaño de la pantalla o aplicación en la que te encuentres.

6.2.6 Descripción de MD5

Conocido simplemente como MD5, es una de las maneras más rápidas y más simples para aumentar la seguridad de los archivos y los mensajes que envía y transferencia.

6.2.7 Descripción del editor de texto sublime Text

Sublime Text es un editor de código multiplataforma, ligero y con pocas concesiones a las florituras. Es una herramienta concebida para programar sin distracciones. Su interfaz de color oscuro y la riqueza de coloreado de la sintaxis, centra nuestra atención completamente.

Sublime Text permite tener varios documentos abiertos mediante pestañas, e incluso emplear varios paneles para aquellos que utilicen más de un monitor. Dispone de modo de pantalla completa, para aprovechar al máximo el espacio visual disponible de la pantalla.

¹⁸ El diseño web adaptable o adaptativo, conocido por las siglas RWD (del inglés, Responsive Web Design) es una filosofía de diseño y desarrollo cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visualizarla. **Fuente especificada no válida.**

¹⁹ La computación grid es una tecnología innovadora que permite utilizar de forma coordinada todo tipo de recursos (entre ellos cómputo, almacenamiento y aplicaciones específicas) que no están sujetos a un control centralizado. **Fuente especificada no válida.**

6.2.8 Descripción de PHP

Es un lenguaje de programación de uso general, de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

Fue uno de los primeros lenguajes de programación del lado de servidor que se podía incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página web resultante.

PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comando que puede ser usada en aplicaciones graficas independientes. Puede ser usada en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo.

6.2.8.1 Comunicación entre el cliente y el servidor

- Escribimos en la barra del web browser (navegador), la dirección y el archivo a solicitar. El navegador envía el mensaje a través de internet a la computadora llamada solicitando la página o archivo “ejemplo.php”.
- El web server (servidor) recibe el mensaje y al ver que la extensión es “PHP” solicita al intérprete de PHP (que es otro programa que se ejecuta en el servidor web) que le envíe el archivo.
- El intérprete de PHP lee desde el disco el archivo “ejemplo.php”.
- El intérprete PHP ejecuta los comandos contenidos en el archivo y eventualmente se comunica con un gestor de base de datos (ejemplo de ello MySQL, Oracle, SQL server, etc.).
- Luego de ejecutar el programa contenido en el archivo, envía este al servidor web.
- El servidor web envía la página al cliente que la había solicitado.
- El navegador muestra en la pantalla el archivo que envió el servidor web.

6.2.8.2 Características

- Es un lenguaje multiplataforma.
- Completamente orientada al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL²⁰.
- Es libre por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables, aunque sus variables se pueden evaluar también por el tipo que este manejando el tiempo de ejecución.

6.2.9 Descripción de Ajax

Ajax, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado (Scripting Language) en el que normalmente se efectúan las

²⁰ PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. **Fuente especificada no válida.**

funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest²¹, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono este formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores, dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

6.2.10 JQuery

JQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC. JQuery es la biblioteca de JavaScript más utilizada.

JQuery es software libre y de código abierto, posee doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privados. JQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

²¹ HTTP :: Request es una clase que encapsula las solicitudes HTTP de estilo, que consiste en una línea de petición, algunas cabeceras, y un cuerpo contenido. **Fuente especificada no válida.**

7. DISEÑO METODOLOGICO

EL COLEGIO Calasanz se encuentre ubicado en la ciudad de León, Nicaragua es una institución educativa, católico y privado. Para el buen funcionamiento del colegio, se debe manejar toda la información referente a los estudiantes desde el momento que llegan a matricularse al colegio hasta el transcurso de ellos en el colegio. Algunos de estos procesos se llevan digitales otros manualmente.

Para estos casos conviene usar un sistema computarizado que pueda hacer más fácil y efectiva la realización de las tareas que deben hacerse en el área de registro académico.

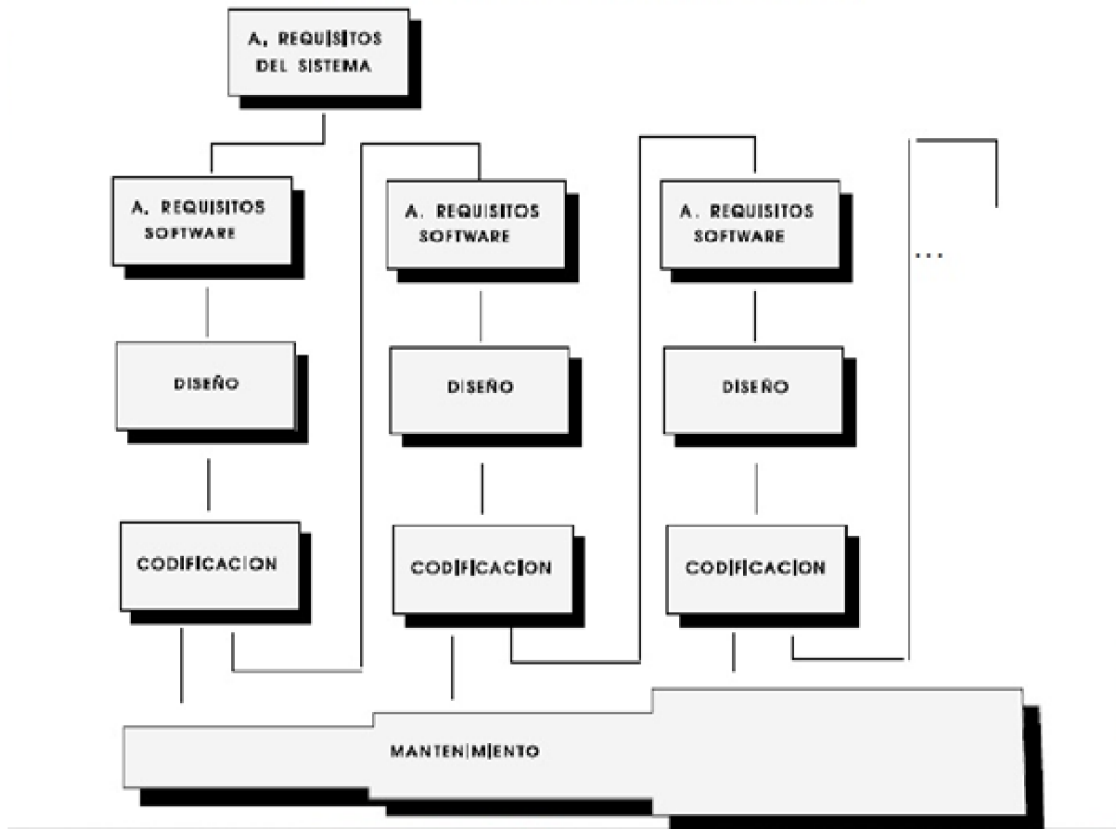
Para realizar el presente trabajo primero se necesita conocer cómo funciona el área de registro académico, así como también cada uno de los subprocesos. Además de eso se necesita saber qué información manipulan y cómo la procesan.

De esta manera tendremos una idea clara de cómo automatizar estos procesos. Una vez que tengamos claro el funcionamiento de esta área y la información con la que se trabaja, se procederá realizar un documento de **Especificación de requisitos**²². El cual será una guía a la hora de desarrollar la aplicación.

Luego se procederá diseñar las interfaces tomando como referencia la especificación de requisito. A partir de las interfaces se iniciará con la codificación de aplicación. El desarrollo de la aplicación se realizará siguiendo el modelo incremental, el paradigma del ciclo de vida abarca las siguientes actividades.

²²La especificación de requisitos de software (ERS), es una descripción completa del comportamiento del sistema que se va a desarrollar. Incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software. (<http://definicion.de/excel/>, 2015)

MODELO INCREMENTAL



El modelo incremental fue propuesto por Harlan Mills en el año 1980. Es un proceso de desarrollo de software creado en respuesta a las debilidades del modelo tradicional de cascada. Surgió el enfoque incremental de desarrollo como una forma de reducir la repetición del trabajo en el proceso de desarrollo y dar oportunidad de retrasar la toma de decisiones en los requisitos hasta adquirir experiencia con el sistema.

Básicamente este modelo de desarrollo, que no es más que un conjunto de tareas agrupadas en pequeñas etapas repetitivas (iteraciones), es uno de los más utilizados en los últimos tiempos ya que, como se relaciona con novedosas estrategias de desarrollo de software y una programación extrema es empleado en metodologías diversas.

El modelo consta de diversas etapas de desarrollo en cada incremento, las cuales inician con el análisis y finalizan con la instauración y aprobación del sistema.

7.1 Ciclo de vida

7.1.1 Etapa de inicialización

Se crea una versión del sistema. La meta de esta etapa es crear un producto con el que el usuario pueda interactuar, y por ende retroalimentar el proceso. Debe ofrecer una muestra de lo aspectos claves del problema y proveer una solución lo suficientemente simple para ser comprendida e implementada fácilmente. Para guiar el proceso de iteración se crea una lista de control de proyecto, que contiene un historial de todas las tareas que necesitan ser realizadas. Incluye cosas como nuevas funcionalidades para ser implementadas y áreas de rediseño de la solución ya existente. Esta lista de control se realiza periódica y constantemente como resultado de la fase de análisis.

7.1.2 Etapa de iteración

Esta etapa involucra el rediseño e implementación de una tarea de la lista de control del proyecto, y el análisis de la versión más reciente del sistema. La meta del diseño e implementación de cualquier iteración es ser simple, directa y modular, para soportar el rediseño de la etapa o como una tarea añadida a la lista de control de proyecto. El código puede, en ciertos casos representar la mayor fuente de documentación del sistema. El análisis de una iteración se basa en la retroalimentación del usuario y el análisis de las funcionalidades disponibles del programa. Involucra el análisis de la estructura, modularidad, usabilidad, confiabilidad, eficiencia y eficacia (alcanzar las metas).

7.1.2.1 Características

- Se evitan proyectos largos y se entrega “algo de valor” a los usuarios con cierta frecuencia.
- Difícil de aplicar a los sistemas transaccionales que tienden a ser integrados ya a operar como un todo.
- El resultado puede ser muy positivo.
- Con un paradigma incremental que reduce el tiempo de desarrollo inicial ya que se implementa la funcionalidad parcial.
- El modelo proporciona todas las ventajas del modelo en cascada realimentando, reduciendo sus desventajas solo al ámbito de cada incremento.
- Resulta más sencillo acomodar cambios al acortar el tamaño de los incrementos.

8. Especificación de requisitos

8.1 Introducción

Este documento es la Especificación de Requisito Software (ERS) para el Desarrollo de una aplicación web de para la gestión de registro académico del colegio Calasanz –León, utilizando el framework Laravel de PHP, en el periodo enero – julio 2015. Todo su contenido ha sido elaborado en colaboración de los integrantes del grupo y tutor. Esta especificación se ha estructurado inspirándose en las directrices dadas por el estándar “IEEE Recommended Practice for Software Requirement Specifications en su estándar 830 [IEEE, 1998]”.

8.2 Propósito

El documento va dirigido al equipo administrativo del colegio Calasanz. Este documento será el canal de comunicación entre las partes implicadas, siendo confeccionado por los miembros del equipo desarrollador y las autoridades del colegio. Esta especificación está sujeta a revisiones por el grupo de trabajo que se recogerán por medio de sucesivas versiones del documento, hasta su aprobación por la totalidad del grupo. Una vez aprobado servirá de base al equipo de desarrollo para la construcción del nuevo sistema.

8.3 Ámbito del Sistema

En la actualidad, el colegio Calasanz cuenta con un sistema de registro académico, el cual no cubre todas las necesidades del colegio referente al manejo de la información académica con la que cuenta, ya que fue creado en años anteriores y no se le ha dado mantenimiento.

La aplicación web para la gestión de registros académicos del colegio Calasanz, tiene como objetivo principal mitigar las necesidades que presenta el sistema que actualmente manejan y brindar seguridad a la información que se almacenara. También se espera alcanzar con esta aplicación la eficacia y rapidez a la hora de manipular los datos, dándoles una respuesta positiva a la institución.

8.4 Definiciones y Acrónimos

8.4.1 Definiciones

Interfaz	Parte del programa informático que permite el flujo de información entre varias aplicaciones o entre el propio programa y el usuario.
Navegador	Aplicación software que permite al usuario recuperar y visualizar documentos de hipertexto.
Portal Web	Página de inicio que permite el acceso a las distintas secciones de un sitio web.
Reportes de notas	Corresponde a los certificados de las calificaciones de los estudiantes.

8.4.2 Acrónimos

ERS	Especificación de requisitos Software
RFXXX	El estándar seguido para la especificación del identificador de cada requisito funcional será de la siguiente manera: R = Requisito F = Funcional XXX = secuencia de tres dígitos que servirá para la enumeración de cada requisito.
RNFXXX	El estándar seguido para la especificación del identificador de cada requisito no funcional será de la siguiente manera: R = Requisito NF = No Funcional XXX = secuencia de tres dígitos que servirá para la enumeración de cada requisito.

8.5 Referencias

- IEEE Recomendad Practices for Software Requierements especification ANSI/IEEE 830 1998.
- Transparencias de la asignatura “Ingeniería del Software I”.
- Apuntes de clase de la asignatura “Ingeniería del Software I”.

8.6 Visión General del Documento

Este documento consta de 4 secciones. Esta sección es la introducción y proporciona una visión general del ERS. En la sección dos se da una descripción general del sistema, con el fin de conocer las principales funciones que debe realizar, los datos asociados y los factores, restricciones, supuestos y dependencias que afectan al desarrollo, sin entrar en excesivos detalles. En la sección tres se definen detalladamente los requisitos que debe satisfacer la aplicación. En la sección cuatro se proporcionará un índice para tener acceso rápido a la documentación presentada en la ERS.

9. Descripción General

Dentro de esta sección se presenta una descripción de alto nivel de la aplicación, definiendo las funciones a realizarse, así como aquellas restricciones generales que afectan al mismo.

9.1 Perspectivas del producto

La aplicación interaccionará con dos productos software. Dichos productos serán comerciales y por lo tanto adquiridos y no desarrollados. Estos productos serán los siguientes:

- **Un servidor web**, que permitirá el acceso a los servicios del sistema. No existe ninguna restricción acerca de que servidor concreto utilizar.
- **Gestor de base de datos MYSQL.**

9.2 Funciones del producto

La aplicación permitirá realizar lo siguiente:

- Editar contraseña del administrador.
- Registrar matrícula.
- Editar matrícula.
- Eliminar matrícula.
- Registrar profesores.
- Editar registros de profesores.
- Eliminar profesores.
- Registrar materia.
- Eliminar materia.
- Asignar materia a un profesor.
- Ver notas.
- Ingresar notas.
- Editar notas.
- Generar reportes de notas.

9.3 Características de los Usuarios

El usuario de ésta aplicación estará familiarizado con el uso de aplicaciones. Es por ello que la aplicación ha de ser gráfica. Con una interfaz sencilla e intuitiva, que no exija gran tiempo para su aprendizaje, a la vez que sea lo suficientemente potente para que el usuario pueda hacer aquello que necesita.

Usuario administrador

Este tipo de usuario se encargará de la gestión de la base de datos de la aplicación; es decir, registrará asignaturas, año de curso, notas, etc. así como las modificaciones necesarias. En general tiene acceso a todo el sistema y podrá realizar todo tipo de procesos.

9.4 Restricciones

Al tratarse de una aplicación web, se requiere de un ordenador con un navegador convencional.

10. Requisitos específicos

Las funciones de la aplicación están orientadas a garantizar la seguridad de control de proceso de registro académico, cumpliendo con todos los procesos requeridos. A continuación, se describen las funciones del producto usando la alternativa de descripción de funciones por usuarios de la aplicación.

- RF001** Editar contraseña.
- RF002** Registrar matrícula.
- RF003** Editar matrícula.
- RF004** Eliminar matrícula.
- RF005** Registrar profesores.
- RF006** Editar registros de un profesor.
- RF007** Eliminar un profesor.
- RF008** Registrar asignatura.
- RF009** Editar asignatura.
- RF010** Eliminar asignatura.
- RF011** Asignar asignatura a un profesor.
- RF012** Ver notas.
- RF013** Registrar notas.
- RF014** Editar notas.
- RF015** Generar reporte de notas.

10.1 Requisitos comunes de las interfaces

10.1.1 Interfaces de usuario

Sera únicamente visible para el administrador, quien será la única persona encargada de administrar la aplicación. Las interfaces de usuario están relacionadas con las pantallas, ventanas (formularios) que debe manipular el usuario para realizar una operación determinada.

10.1.2 Interfaces Hardware

Para desplazarse a través de las diferentes opciones y características será necesario utilizar el ratón (mouse) y teclado.

Servidor web y servidor de base de datos, ambos son necesarios para que la aplicación funcione correctamente.

10.1.3 Interfaces Software

La aplicación funcionará en cualquier máquina con un navegador web.

10.1.4 Interfaces del Sistema

Al tratarse de una aplicación web, se podrá visualizar sobre cualquier sistema operativo.

10.1.5 Interfaces de comunicación

Esta aplicación deberá guardar registros en una base de datos MYSQL, con la ayuda del lenguaje de programación PHP, el cual interactúa en el lado del servidor.

10.2 Requisitos funcionales

A continuación, se describen con más detalles estas tareas y cómo serán soportadas por el sistema:

10.2.1 Requisito funcional 1

Número del requisito	RF001
Nombre del requisito	Editar contraseña
Tipo	Requisito
Fuente del requisito	Tabla contraseña

Introducción

La aplicación constará con un formulario en el cual el administrador deberá ingresar la contraseña actual, luego la nueva contraseña y posteriormente repetir la nueva contraseña.

Entrada

El sistema tendrá una ventana que permita editar los datos.

Proceso

Los datos insertados en el formulario se enviarán a la base de datos.

Salida

Un mensaje que indica que la contraseña se ha modificado correctamente.

10.2.2 Requisito funcional 2

Número del requisito	RF002
Nombre del requisito	Registrar matricula
Tipo	Requisito
Fuente del requisito	Tabla matricula

Introducción

La aplicación constará con un formulario en el cual el administrador podrá ingresar todos los datos requeridos del estudiante para realizar una matrícula.

Entrada

La aplicación tendrá una ventana que permita la inserción de los datos.

Proceso

Los datos insertados se enviarán a la base de datos.

Salida

Un mensaje que indica que la matrícula se realizó exitosamente.

10.2.3 Requisito funcional 3

Número del requisito	RF003
Nombre del requisito	Editar matrícula
Tipo	Requisito
Fuente del requisito	Tabla matrícula

Introducción

La aplicación constará con un formulario en el cual el administrador podrá editar los datos necesarios de una matrícula.

Entrada

La aplicación tendrá una ventana que permita la edición de los datos.

Proceso

Los datos editados se enviarán a la base de datos.

Salida

Un mensaje que indica que la edición se realizó exitosamente.

10.2.4 Requisito funcional 4

Número del requisito	RF004
Nombre del requisito	Eliminar matrícula
Tipo	Requisito
Fuente del requisito	Tabla matrícula

Introducción

La aplicación constará con la opción de eliminar una matrícula. Esto será necesario en caso de que el estudiante se halla retirado y no vaya a continuar durante ese año escolar.

Entrada

El administrador seleccionará la matricula a eliminar.

Proceso

El id de la matricula a eliminar será enviada a la base de datos.

Salida

Un mensaje que indica que la matrícula se eliminó exitosamente.

10.2.5 Requisito funcional 5

Número del requisito	RF005
Nombre del requisito	Registrar profesores
Tipo	Requisito
Fuente del requisito	Tabla profesores

Introducción

La aplicación constará con un formulario en el cual el administrador podrá ingresar todos los datos requeridos del profesor, cuando hay una nueva contratación de docente.

Entrada

La aplicación tendrá una ventana que permita la inserción de los datos del profesor.

Proceso

Los datos insertados en el formulario se enviarán a la base de datos.

Salida

Un mensaje que indica que el registro se realizó exitosamente.

10.2.6 Requisito funcional 6

Número del requisito	RF006
Nombre del requisito	Editar registros de un profesor
Tipo	Requisito
Fuente del requisito	Tabla profesores

Introducción

La aplicación constará con la opción de poder editar los datos necesarios de un profesor en específico, esto en caso de que algunos datos del mismo no se hallan insertados correctamente.

Entrada

La aplicación tendrá una ventana que permita la edición de los datos del profesor.

Proceso

Los datos editados se enviarán a la base de datos.

Salida

Un mensaje que indica que la edición se realizó exitosamente.

10.2.7 Requisito funcional 7

Número del requisito	RF007
Nombre del requisito	Eliminar un profesor
Tipo	Requisito
Fuente del requisito	Tabla profesores

Introducción

La aplicación constará con la opción de dar de baja a un maestro.

Entrada

El administrador se encargará de seleccionar el maestro que será dado de baja.

Proceso

El id del maestro será enviado a la base de datos para ser eliminado.

Salida

Un mensaje que indica que la operación se realizó exitosamente.

10.2.8 Requisito funcional 8

Número del requisito	RF008
Nombre del requisito	Registrar asignatura
Tipo	Requisito
Fuente del requisito	Tabla asignatura

Introducción

La aplicación constará con un formulario en el cual el administrador podrá ingresar todos los datos solicitados referentes a una nueva asignatura, en caso que se vaya a impartir una nueva.

Entrada

La aplicación tendrá un formulario que solicite llenar los datos referentes a la materia.

Proceso

Los datos insertados en el formulario se enviarán a la base de datos.

Salida

Un mensaje que indica que la nueva asignatura se creó exitosamente.

10.2.9 Requisito funcional 9

Número del requisito	RF009
Nombre del requisito	Editar asignatura
Tipo	Requisito
Fuente del requisito	Tabla asignatura

Introducción

La aplicación constará con la opción de editar una asignatura, como el nombre, por ejemplo.

Entrada

El administrador seleccionará la asignatura a editar.

Proceso

Los datos editados de la asignatura se enviarán a la base de datos.

Salida

Un mensaje que indica que la edición se creó exitosamente.

10.2.10 Requisito funcional 10

Número del requisito	RF010
Nombre del requisito	Eliminar asignatura
Tipo	Requisito
Fuente del requisito	Tabla asignatura

Introducción

La aplicación constará con la opción de eliminar una asignatura, en caso de que la misma deje de impartirse.

Entrada

El administrador seleccionará la asignatura a eliminar.

Proceso

El id de la asignatura a eliminar será enviada a la base de datos.

Salida

Un mensaje que indica que la asignatura se eliminó exitosamente.

10.2.11 Requisito funcional 11

Número del requisito	RF011
Nombre del requisito	Asignar asignatura a un profesor
Tipo	Requisito
Fuente del requisito	Tabla asignatura y profesor

Introducción

La aplicación constará con la opción de asignar una respectiva asignatura al maestro que se encargará de impartirla.

Entrada

El administrador se encargará de asociar una asignatura a un maestro.

Proceso

Los datos asociados se enviarán a la base de datos.

Salida

Un mensaje que indica que la operación se produjo exitosamente.

10.2.12 Requisito funcional 12

Numero de requisito	RF012
Nombre del requisito	Ver notas
Tipo	Requisito
Fuente del requisito	Tabla de notas

Introducción

La aplicación debe mostrar un reporte de las notas de los estudiantes, ya sea por grupo o de manera individual.

Entrada

El usuario seleccionará el id del estudiante del que desea ver la nota, y en caso de ser un grupo, seleccionará el grado y la sección del grupo a visualizar.

Proceso

Este valor se enviará a la base de datos y a través de una consulta devolverá la nota que se registró.

Salida

Una interfaz mostrando los datos devueltos por la base de datos.

10.2.13 Requisito funcional 13

Número del requisito	RF013
Nombre del requisito	Registrar notas
Tipo	Requisito
Fuente del requisito	Tabla de notas

Introducción

Las notas de los estudiantes serán registradas mediante el uso de un archivo en EXCEL.

Entrada

La aplicación obtendrá las notas de los estudiantes desde un archivo en EXCEL, anteriormente el administrador seleccionará el grado, la sección, la materia y el año lectivo de la cual pertenecen las notas.

Proceso

La aplicación primero verifica si los datos seleccionados (grado, sección, materia y año lectivo) por el administrador a la hora de subir las notas coinciden con los datos del archivo EXCEL suministrados por el maestro respectivo. En caso afirmativo los datos serán volcados a la base de datos, en caso contrario, mostrará una alerta y los datos no serán guardados.

Salida

Un mensaje que indica que la nota se ha insertado correctamente.

10.2.14 Requisito funcional 14

Número del requisito	RF014
Nombre del requisito	Editar notas
Tipo	Requisito
Fuente de requisito	Tabla notas

Introducción

La aplicación mostrará una ventana que permita cambiar el valor existente por uno nuevo.

Entrada

La aplicación tendrá una ventana que permita cambiar el valor numérico.

Proceso

Los datos insertados en el formulario se enviarán a la base de datos.

Salida

Se mostrará un mensaje de que la operación fue exitosa.

10.2.15 Requisito funcional 15

Numero de requisito	RF015
Nombre del requisito	Generar reporte de notas
Tipo	Requisito
Fuente del requisito	Tabla de notas

Introducción

La aplicación debe mostrar un reporte de las notas de los estudiantes, ya sea por grupo o de manera individual.

Entrada

El usuario seleccionará el id del estudiante del que desea ver la nota, y en caso de ser un grupo, seleccionará el grado y la sección del grupo a visualizar.

Proceso

Este valor se enviará a la base de datos y a través de una consulta devolverá la nota que se registró. Además de visualizar las notas, el sistema será capaz de generar un reporte de las notas en un archivo en PDF, listo para su impresión en caso de que el administrador así lo desee.

Salida

Un archivo en PDF con las notas, que permitirá la impresión del documento, en caso que se quiera en físico.

10.3 Requisitos de Rendimiento

Será necesario disponer de un ordenador con un navegador web y un servidor MySQL (preferiblemente XAMPP, ya que dispone de MySQL y Apache).

10.3.1 Recursos disponibles

Para la realización de nuestro sistema web a nivel hardware requerimos de ciertas herramientas mencionadas a continuación.

- PC: AMD Dual-Core A6-4455M ~2.1GHz.
- RAM: 8GB.
- Disco duro: 500 GB HDD.
- Sistema Operativo: Windows 10 Pro 64 bits.

10.4 Atributos de la aplicación

10.4.1 Seguridad

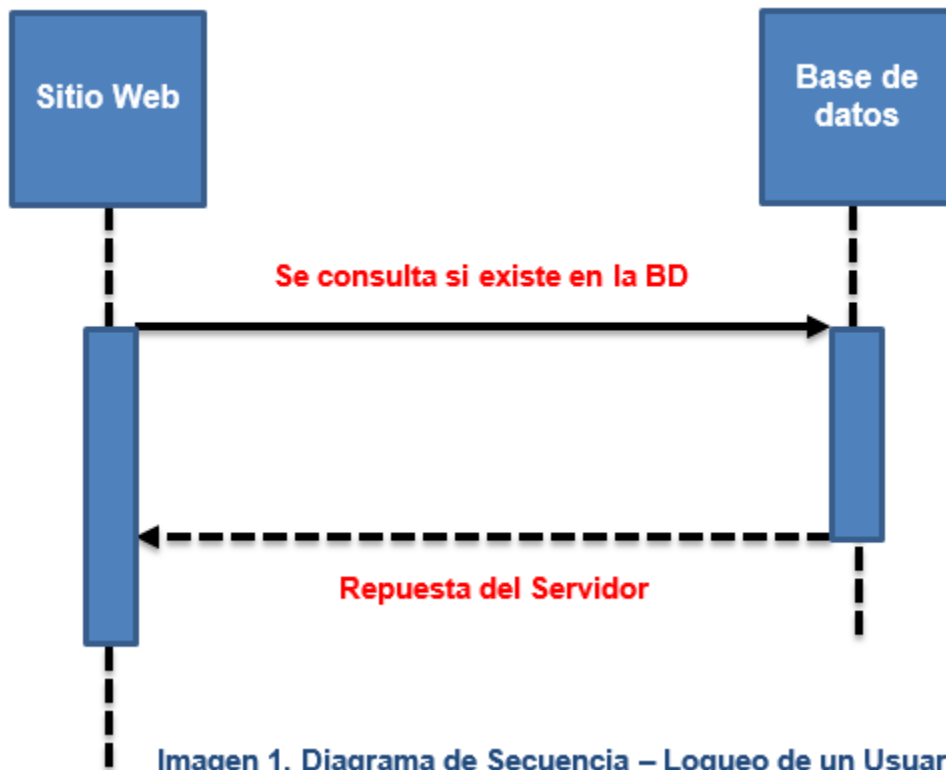
La aplicación tendrá un solo tipo de usuario, para mayor seguridad:

Administrador: Será el encargado de controlar todas las funciones que se realizan y solo él dispondrá de un usuario y contraseña para acceder a la aplicación.

11. Diagramas de secuencia

11.1 Diagrama de Secuencia 1: Inicio de sesión de un usuario.

Al iniciar la aplicación, el administrador debe iniciar sesión, introduciendo su usuario y contraseña. Una vez insertado los datos en el formulario estos se validan para verificar si son correctos y si el usuario existe o no en la base de datos. Dependiendo de la información que el sistema gestor de base de datos retorne (True o False), la aplicación permitirá o denegará el acceso al usuario solicitante.



11.2. Diagrama de secuencia 2: Registrar notas.

Cuando el administrador selecciona la opción de registrar notas, se llenan los datos que se piden en el formulario, uno de estos es seleccionar el archivo a subir, luego se abre una nueva ventana para localizar el archivo, se selecciona el archivo y los datos se envían a la base de datos de la aplicación. La base de datos enviará una notificación que indique si la operación fue exitosa (los datos fueron ingresados) o si la operación culminó con error, en éste último caso habrá que ingresar nuevamente los datos.

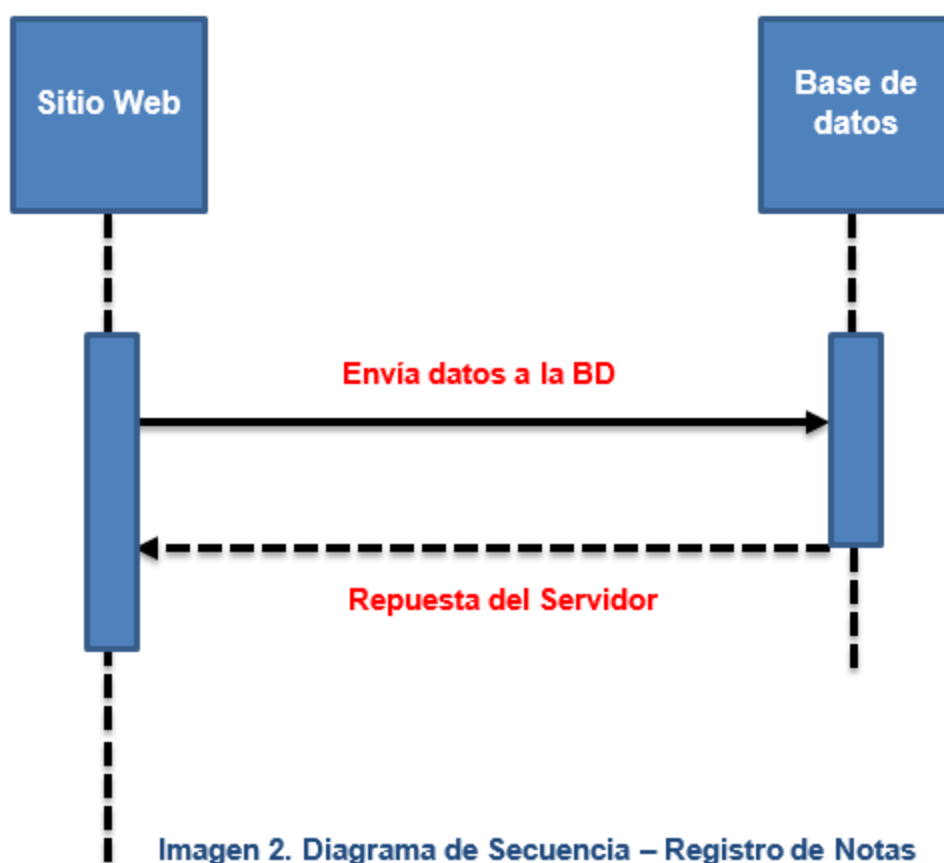


Imagen 2. Diagrama de Secuencia – Registro de Notas

11.3 Diagrama de secuencia 3: Registrar matricula.

Los datos del estudiante a matricular son ingresados en un formulario y luego enviados a la base de datos. Nuevamente el gestor de base de datos se encargará de enviar una notificación a la aplicación, dependiendo si la operación se realizó con éxito (los datos fueron insertados exitosamente) o en caso contrario una notificación de error.

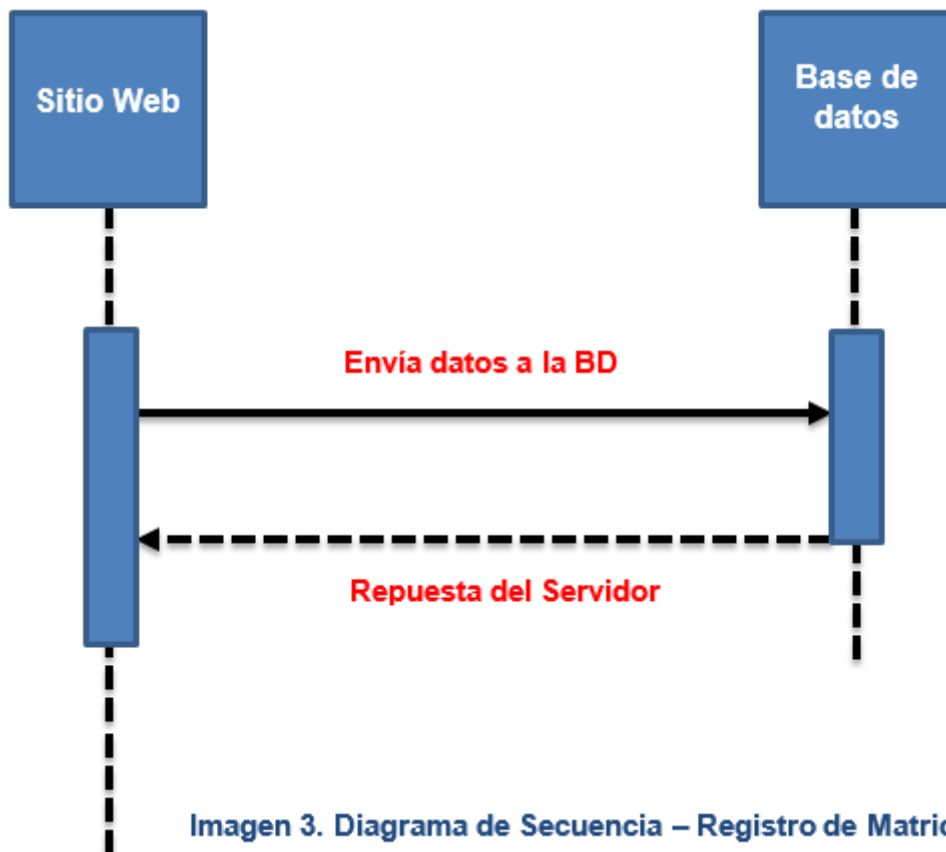


Imagen 3. Diagrama de Secuencia – Registro de Matriculas

11.4 Diagrama de secuencia 4: Registrar profesores.

El administrador será el encargado de registrar a un nuevo docente cuando las circunstancias del centro educativo lo ameriten. Llena los campos con los datos requeridos en el formulario y estos datos son almacenados en la base de datos. El servidor responderá indicando si tuvo éxito en la inserción o no.

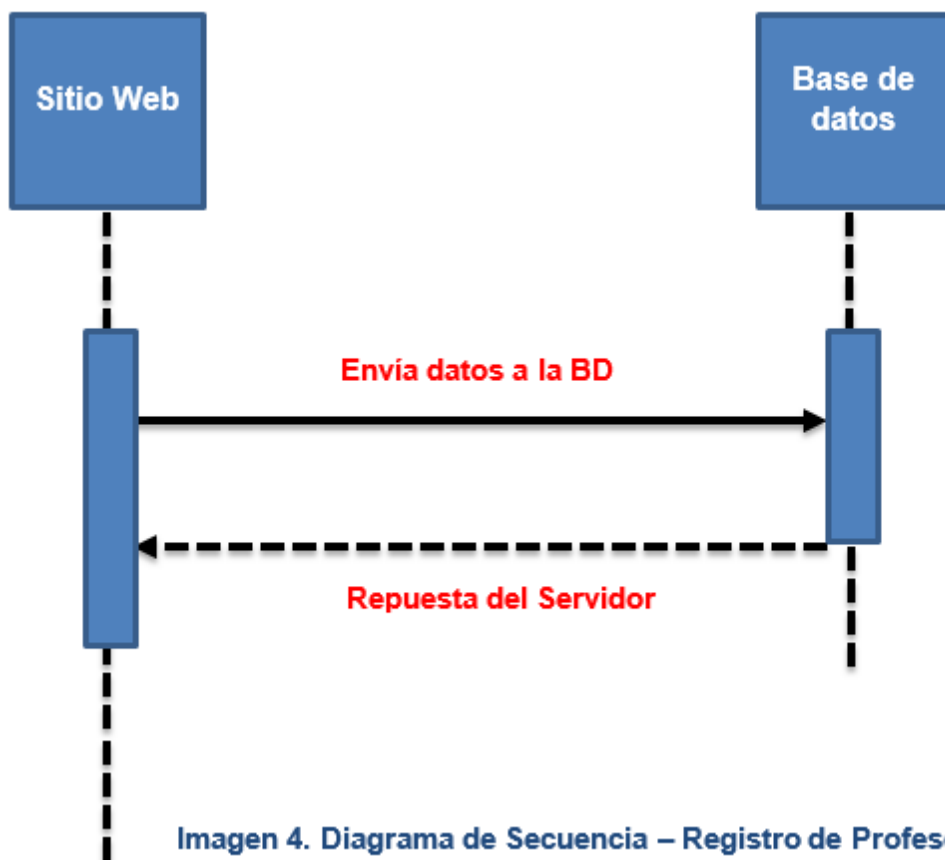


Imagen 4. Diagrama de Secuencia – Registro de Profesores

11.5 Diagrama de secuencia 5: Registrar materia.

Este proceso se llevará a cabo cuando se quiera registrar una nueva asignatura que no está registrada en la aplicación. En el formulario correspondiente se llenan los datos solicitados y éstos se envían a la base datos para su almacenamiento, el servidor a su vez notificara a la aplicación el estado final de la operación (exitoso o error).

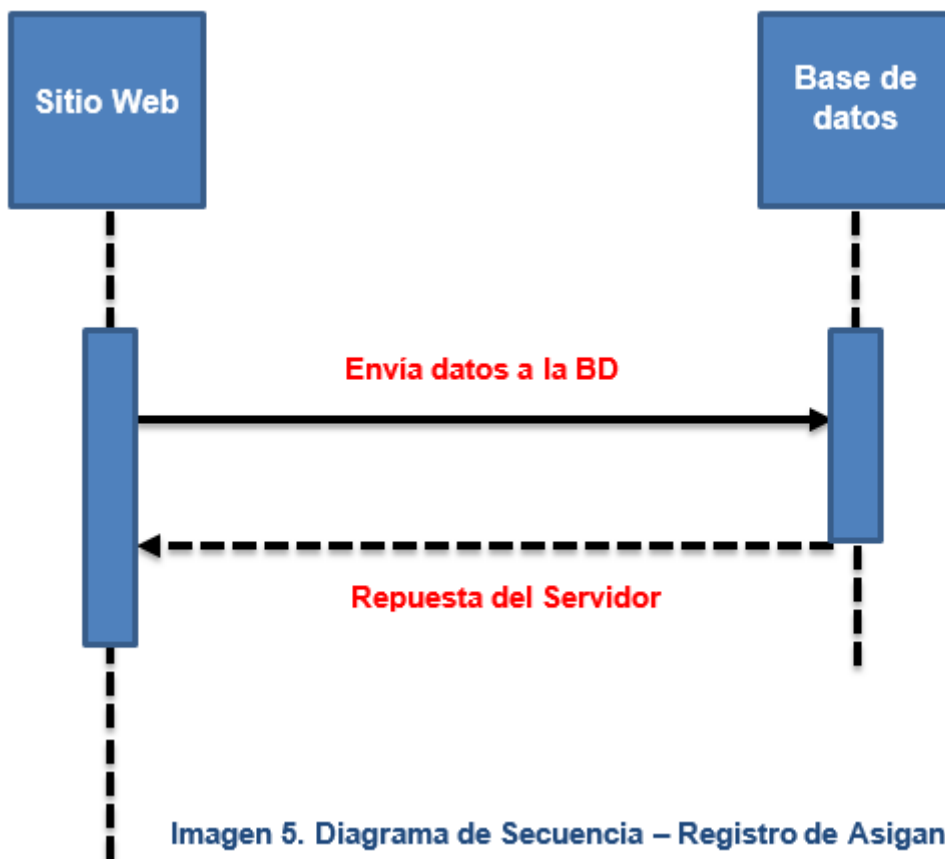
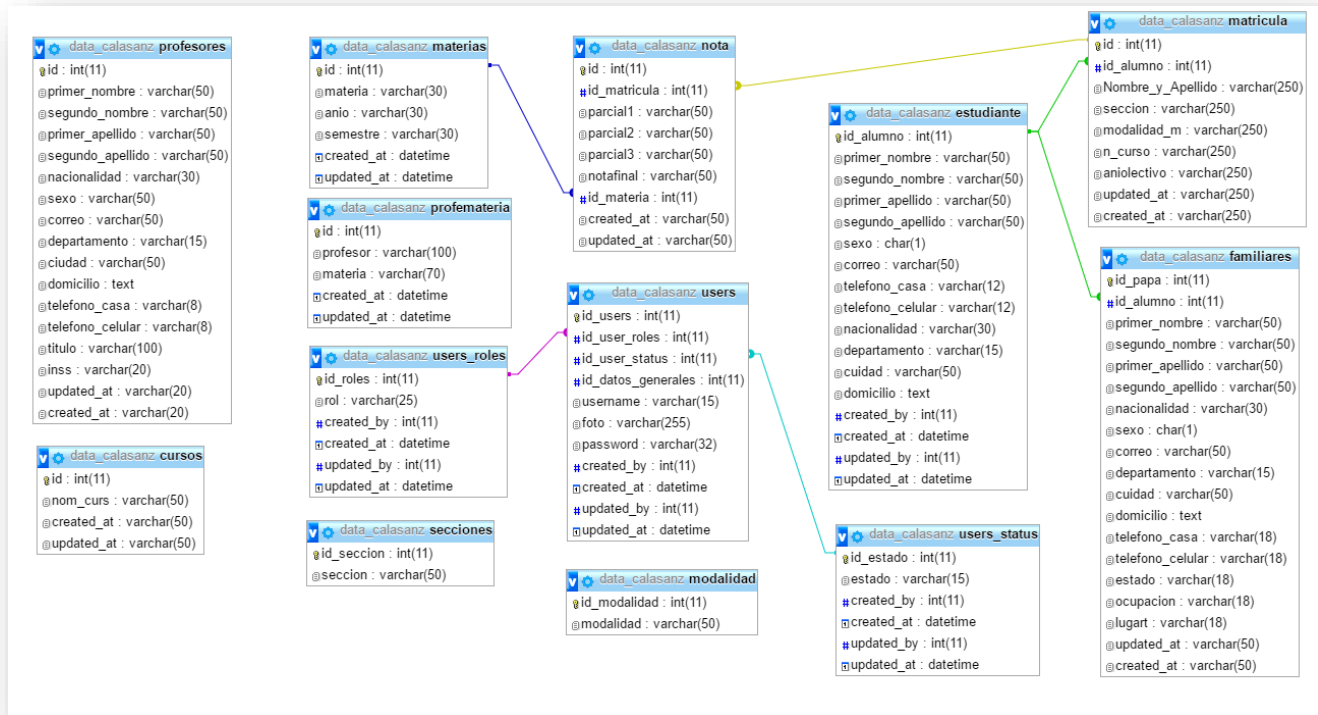


Imagen 5. Diagrama de Secuencia – Registro de Asignaturas

12. ANEXOS

12.1 Modelo Entidad – Relación

En este apartado se muestra el modelo entidad – relación de la base de datos de la aplicación. Dicho modelo refleja tanto las tablas que conforman la base de datos, como la relación entre ellas.



12.2 Diseño de la interfaz

12.2.1 Pantalla principal de la aplicación web.

Al iniciar la aplicación se mostrará la página principal del proyecto, en este instante la aplicación está protegido de tal manera que solo pueda acceder un administrador mediante un Usuario y Contraseña, para lo cual será necesario un inicio de sesión.



12.2.2 Ingreso del Usuario y Contraseña.

En este instante el administrador necesita iniciar sesión como requisito indispensable para poder ingresar a la aplicación. La aplicación verificara si ese usuario existe en la base de datos, en caso afirmativo, dará la Bienvenida al usuario y permitirá acceder a todas las funciones del registro académico; en caso contrario, simplemente enviara un mensaje a la pantalla de que el usuario no existe en la base de datos.



The screenshot shows a login form with a dark header containing the text "Acceder al sistema". Below the header are two input fields: the first is labeled "Ingrese el nombre de usuario" and the second is labeled "Ingrese la contraseña". At the bottom of the form is a blue button labeled "Acceder!".

12.2.2.1 Usuario y/o Contraseña incorrectos.



The screenshot shows an error message page with the title "Error de Inicio de Sesión". Below the title is the text "Usuario o contraseña incorrecta..". At the bottom of the page is a blue button labeled "Regresar".

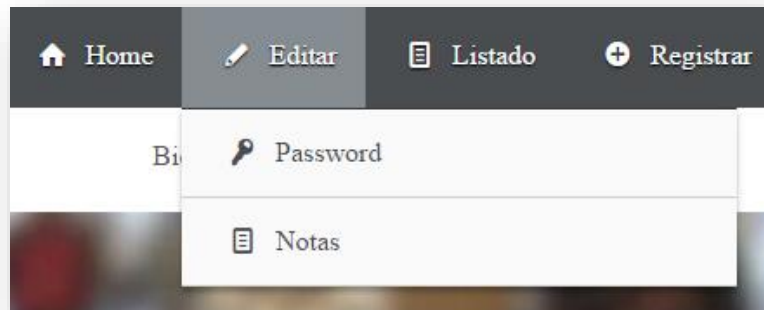
12.2.2.1 Usuario y Contraseña correctos.



Al ingresar tanto el **Usuario** como la **Contraseña** de forma correcta, la aplicación dará la Bienvenida al Usuario y lo redirigirá a la pantalla de Inicio, pero en esta ocasión ya se encontrarán habilitadas todas las funciones necesarias para el registro académico de la institución propietaria.

12.2.3 Edición de contraseña.

Una vez dentro de la aplicación, el administrador dispondrá de la opción de modificar su propia contraseña. Esta opción le permitirá sustituirla por una más segura o más fácil de recordar.

A screenshot of a mobile application screen titled 'Editar la contraseña'. The screen has a dark grey header with the title in white. Below the header, there are three white input fields with rounded corners. The first field contains the placeholder text 'Ingrese la contraseña actual'. The second field contains 'Ingrese la contraseña nueva'. The third field contains 'Re-Ingrese la contraseña nueva'. At the bottom of the form is a blue button with white text that says 'Cambiar contraseña'.

Una vez realizado este proceso, el cambio es instantáneo, lo que significa que la próxima vez tendrá que hacer uso de la nueva contraseña.

12.2.4 Listado de alumnos existentes

Algo necesario para la inserción de notas es que la aplicación misma genere un documento con la lista actualizada de estudiantes, precisamente es lo que se puede ver a continuación:



The screenshot shows a web application interface with a dark navigation bar at the top containing icons and labels for 'Home', 'Editar', 'Listado', 'Registrar', and a help icon. Below the navigation bar, the text 'Modulo para crear los Listados de Alumnos' is displayed. The main content area features a white modal box with a dark header titled 'Generar Listado de Alumnos'. Inside the modal, there are three dropdown menus: 'Seleccione el Curso', 'Seleccione la Sección', and 'Seleccione el año'. Below these menus is a blue button labeled 'Generar Listado'.

En este caso, primero se tendrá que seleccionar el Curso, la Sección y el año del cual se necesita obtener el listado, luego al darle clic en el botón “Generar Listado”, la aplicación generará un archivo en Excel, con todos los datos de los estudiantes que pertenecen al Grado, Sección y año antes seleccionados.

12.2.5 Pestaña de “Registrar”



Desde esta opción en la página principal de la aplicación, podemos realizar varias operaciones como:

12.2.5.1 Registrar Matricula

Sera necesario cuando se desee agregar un nuevo estudiante a la base de datos o cuando se desee matricular a un nuevo año escolar.

Datos generales del alumno

Primer nombre	Segundo nombre
Primer apellido	Segundo apellido
Correo	
Telefono convencional	Telefono celular
Seleccione Sexo ▼	Nacionalidad
Departamento	Cuidad
Domicilio	
Siguiente	

12.2.5.2 Registrar Profesor

Indispensable cuando se vaya a agregar a un nuevo maestro aun no existente en la base de datos.

Crear Nuevo Profesor

Primer nombre	Segundo nombre
Primer apellido	Segundo apellido
Correo	
Telefono convencional	Telefono celular
Seleccione Sexo ▼	Nacionalidad
Departamento	Cuidad
Domicilio	
Titulo	Inss
Guardar	

12.2.5.3 Registrar Materia

Será necesario cuando la institución vea necesario agregar una nueva asignatura.

Crear Nueva Materia

12.2.5.4 Asignar Materia a Profesor

Una vez registrados todos los maestros que se encargarán de impartir todas las asignaturas, será necesario asociar cada materia a cada profesor que la impartirá.

Asignar Materia a Profesor

Seleccione el Profesor ▼

Seleccione la Materia ▼

Guardar

Asignacion de Materia

La materia se asigno Correctamente

Regresar

12.2.6 Pestaña de “Notas”

En este apartado se podrá tanto visualizar las notas de los estudiantes, como ingresar las nuevas notas de casa parcial.



12.2.6.1 Ver notas

La aplicación dispone de 2 opciones para visualizar notas:

12.2.6.1.1 Búsqueda de un estudiante en particular.

Necesario para realizar una búsqueda individual, se necesitará ingresar el curso y el nombre del estudiante en el campo en donde se solicita. Luego solo necesitaremos presionar el botón “Buscar Estudiante”.

Ingrese los datos

Seleccione el Curso ▼

Escriba apellido o carnet.

Buscar Estudiante

Reporte General



Colegio Calasanz - León

Nombre del Alumno : Jader Efrain Sevilla Centeno

Grado : Undecimo

Año Escolar Lectivo : 2015

Seccion: A

Asignaturas:

Asignaturas	I Parcial	II Parcial	III Parcial
Visual C#	85	90	86
matematica	85	90	86
Administracion deRedes	85	90	86
Visual C#	85	90	86

Generar Reporte

Regresar

Ahora ya podemos ver las notas del estudiante seleccionado, en caso de querer las notas en un formato en físico, solo será necesario que demos clic en el botón “Generar Reporte”, esta opción generará un archivo en formato pdf y desde allí ya podremos mandar a imprimirlo.

12.2.6.1.2 Búsqueda de un grupo de estudiantes.

Ingrese los datos

Seleccione el Curso ▼

Escriba apellido o carnet.

Buscar Estudiante **Reporte General**

Reporte general de notas por materia

Seleccione un curso ▼

Seleccione la sección ▼

Descargar **Regresar**

Esta opción permitirá mostrar todos los estudiantes que pertenezcan a un mismo año y sección y de ser necesario generar un reporte de notas.

12.2.6.2 Ingresar notas



Para poder ingresar una nueva nota, primero será necesario seleccionar la Materia, el Curso y el año lectivo al cual se ingresarán las nuevas notas, además de un archivo de tipo Excel en el cual se encuentran las notas.

A screenshot of a web form titled 'Subir las Notas'. The form has a dark grey header with the title. Below the header, there are four input fields: three dropdown menus labeled 'Seleccione la Materia', 'Seleccione el Curso', and 'Seleccione el Año Lectivo', and a file selection field labeled 'Seleccionar archivo' with the text 'Ningún archivo seleccionado' next to it. At the bottom of the form is a blue button labeled 'Guardar'.

La aplicación se encargará de tomar las notas y vaciarlas a la base de datos de acuerdo a los parámetros que se le hayan introducido.

13. CONCLUSIONES

Una vez analizada la información, diseñada y codificada la aplicación en el presente trabajo se ha llegado a las siguientes conclusiones:

- Se analizó y documentó de manera detallada todo el proceso de análisis, diseño y desarrollo de la aplicación.
- Se rediseño en su totalidad la base de datos, creando una nueva estructura y normalizando cada uno de sus módulos, con el objetivo de obtener una integridad en los datos mejorada y confiable.
- Se desarrolló una nueva aplicación de registro académico en ambiente Web para el colegio Calasanz de la ciudad de León que cumple con toda la funcionalidad del sistema actual.
- Se hicieron mejoras al sistema, incluyendo reportes exclusivos y mejora de la interfaz gráfica de usuario.
- La arquitectura de la aplicación permite ser ejecutada en cualquier computadora cliente que posea algún navegador web.

14. BIBLIOGRAFIA

Referencias

Henry A. Rodriguez C. (junio de 2008). *tesis.ula.ve.*

Obtenido de http://tesis.ula.ve/pregrado/tde_busca/archivo.php?codArchivo=1884

<http://definicion.de/excel/>. (2015).

Obtenido de <http://definicion.de/excel/>

<http://definicion.de/sql/>. (2015).

Obtenido de <http://definicion.de/sql/>

<http://es.slideshare.net/laemoyelemo/era-de-la-informacion-5184737>. (2015).

Obtenido de <http://es.slideshare.net/laemoyelemo/era-de-la-informacion-5184737>

<http://laravel.com/>. (2015).

Obtenido de <http://laravel.com/>

<http://www.w3.org/html/>. (2014).

Obtenido de <http://www.w3.org/html/>

<https://acrobat.adobe.com/la/es/products/about-adobe-pdf.html>. (2015).

Obtenido de <https://acrobat.adobe.com/la/es/products/about-adobe-pdf.html>

www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf. (2015).

Obtenido de www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf

Apache Friends. (2015). www.apachefriends.org. (Copyright)

Recuperado el 26 de marzo de 2015, de <https://www.apachefriends.org/es/index.html>

Obtenido de <https://www.apachefriends.org/es/index.html>

ASP.NET. (2015). <http://www.asp.net/mvc>.

Obtenido de <http://www.asp.net/mvc>

Grid'S. (2015). <http://www.thegridsystem.org/>.

Obtenido de <http://www.thegridsystem.org/>

Henry A. Rodriguez C. (junio de 2008). *tesis.ula.ve.*

Obtenido de http://tesis.ula.ve/pregrado/tde_busca/archivo.php?codArchivo=1884
<http://php.net/>. (2015). <http://php.net/>.

Obtenido de <http://php.net/>
<http://www.md5.net/>. (2015). <http://www.md5.net/>.

Obtenido de <http://www.md5.net/>
<http://www.sublimetext.com/>. (s.f.). <http://www.sublimetext.com/>.

Obtenido de <http://www.sublimetext.com/>
<https://www.apachefriends.org/es/index.html>. (2015).

Obtenido de <https://www.apachefriends.org/es/index.html>
ISO. (1 de Noviembre de 2006). http://www.iso.org/iso/catalogue_detail?csnumber=43485.

Obtenido de http://www.iso.org/iso/catalogue_detail?csnumber=43485
Laravel. (s.f.). <http://laravel.com/docs/5.0/eloquent>.

Obtenido de <http://laravel.com/docs/5.0/eloquent>
Perl. (2015). <https://www.perl.org/>.

Obtenido de <https://www.perl.org/>
PostgreSQL. (2013). http://www.postgresql.org/es/sobre_postgresql.

Obtenido de http://www.postgresql.org/es/sobre_postgresql
Request, H. . (2015). <http://search.cpan.org/~gaas/HTTP-Message-6.06/lib/HTTP/Request.pm>.

Obtenido de <http://search.cpan.org/~gaas/HTTP-Message-6.06/lib/HTTP/Request.pm>
Responsive. (2015). <http://responsivetest.net/#u=http://www.magentothemes.net|1024|768|1>.

Obtenido de <http://responsivetest.net/#u=http://www.magentothemes.net|1024|768|1>
UNAN-León. (2006). <https://www.sigacad.unanleon.edu.ni/>.

Obtenido de <http://www.sigacad.edu.ni>
XML. (23 de Abril de 2015). <http://www.xml.com/>.

Obtenido de <http://www.xml.com/>