

**Análisis, diseño e implementación de prototipo de aplicativo web para la
gestión de notas de la Institución Educativa Agustín Nieto Caballero**

Christhian Julián Gómez Castaño

Duvan Sebastian Florez Guerrero

Universidad Tecnológica de Pereira

Programa de ingeniería de sistemas y computación

Facultad de ingenierías

Pereira - Risaralda

2018

**Análisis, diseño e implementación de prototipo de aplicativo web para la
gestión de notas de la Institución Educativa Agustín Nieto Caballero**

Christhian Julián Gómez Castaño

Duvan Sebastian Florez Guerrero

Proyecto de grado

Para optar por el título de ingeniería de sistemas y computación

Cesar Manuel Castillo Rodríguez

Asesor

Programa de ingeniería de sistemas y computación

Facultad de ingenierías

Pereira - Risaralda

2018

RESUMEN

El presente proyecto se enfoca en el desarrollo de un prototipo de aplicativo web, para la gestión de calificaciones y observaciones de los estudiantes de la institución educativa, Agustín Nieto Caballero que se encuentra ubicada en Risaralda Dosquebradas de Colombia

Durante la ejecución del proyecto se utilizó la metodología ágil XP (Extreme Programming), que fue desarrollada por Kent Beck cuyo propósito es garantizar que los procesos del desarrollo de software sean simples, rápidos y funcionales, para el desarrollo de medianos y pequeños proyectos.

El desarrollo del aplicativo web fue llevado a cabo mediante un framework conocido como Laravel, que se destaca por la simplicidad y rapidez con el que se logra desarrollar software de calidad orientado a la web.

AGRADECIMIENTOS

Al ingeniero César Manuel Castillo Rodríguez, quien nos asesoró y estuvo al tanto del desarrollo del proyecto.

Al Director Fixonder Quiroz de la Institución Educativa Agustín Nieto Caballero por su amabilidad y disposición para el desarrollo del presente proyecto web.

TABLA DE CONTENIDO

1	INTRODUCCIÓN.....	10
2	DEFINICIÓN DEL PROBLEMA.....	10
2.1	DESCRIPCIÓN O PLANTEAMIENTO DEL PROBLEMA	10
3	JUSTIFICACIÓN	11
4	OBJETIVOS.....	11
4.1	OBJETIVO GENERAL	11
4.2	Objetivos específicos.....	11
5	MARCO REFERENCIAL	12
5.1	MARCO CONTEXTUAL	12
5.1.1	RESEÑA HISTÓRICA DE LA INSTITUCIÓN.	12
5.2	MARCO CONCEPTUAL.....	13
6	METODOLOGÍA.....	19
7	DESARROLLO DEL PROYECTO	21
7.1	TOMA DE REQUERIMIENTOS	21
7.1.1	LISTA DE REQUERIMIENTOS	21
7.2	CASOS DE USO	22
7.2.1	ROLES DEL SISTEMA	22
7.2.2	DESCRIPCIÓN DE CASOS DE USO	22
7.2.3	DIAGRAMAS DE CASOS DE USO	37
7.3	DISEÑO DEL MODELO DE DATOS	39
7.3.1	MODELO DE DATOS DEL SISTEMA.....	39
7.3.2	MODELO DE DATOS PARA EL INICIO DE SESION.....	39
7.4	DISEÑO DE PROTOTIPOS DE PANTALLA	40
7.5	IMPLEMENTACIÓN DEL SISTEMA.....	45
7.5.1	MÓDULO DE ADMINISTRADOR	45

7.5.2	MÓDULO DE DOCENTE.....	46
7.5.3	MÓDULO DE ESTUDIANTE.....	46
7.5.4	MÓDULO DE ACUDIENTE.....	47
7.5.5	MÓDULO DE INICIO DE SESIÓN.....	48
7.6	PRUEBAS.....	49
7.6.1	ELEMENTOS DE PRUEBA.....	49
7.6.2	RESULTADOS DE LAS PRUEBAS.....	50
8	CONCLUSIONES.....	51
9	RECOMENDACIONES.....	52
10	BIBLIOGRAFÍA.....	53

Tabla de Ilustraciones

ILUSTRACIÓN 1. DIAGRAMA DE CASOS DE USO ADMINISTRADOR (ELABORACIÓN PROPIA)	37
ILUSTRACIÓN 2. DIAGRAMA DE CASOS DE USO ADMINISTRADOR - CREACIÓN DE USUARIOS (ELABORACIÓN PROPIA)	37
ILUSTRACIÓN 3. DIAGRAMA DE CASOS DE USO DOCENTE (ELABORACIÓN PROPIA)	38
ILUSTRACIÓN 4. DIAGRAMA DE CASOS DE USO - ESTUDIANTE ACUDIENTE (ELABORACIÓN PROPIA)	38
ILUSTRACIÓN 5. DIAGRAMA RELACIONAL - SISTEMA (ELABORACIÓN PROPIA)	39
ILUSTRACIÓN 6. DIAGRAMA RELACIONAL - INICIO DE SESIÓN (ELABORACIÓN PROPIA)	39
ILUSTRACIÓN 7. PROTOTIPO DE PANTALLA -INICIO DE SESIÓN (ELABORACIÓN PROPIA)	40
ILUSTRACIÓN 8. PROTOTIPO DE PANTALLA – CREACIÓN/EDICION DE UN USUARIO (ELABORACIÓN PROPIA)	41
ILUSTRACIÓN 9. PRO TIPO DE PANTALLA - CREACIÓN/EDICIÓN DE UN GRUPO (ELABORACIÓN PROPIA)	41
ILUSTRACIÓN 10. PROTOTIPO DE PANTALLA – GESTIÓN DE ACTIVIDADES Y CALIFICACIONES (ELABORACIÓN PROPIA)	42
ILUSTRACIÓN 11. PROTOTIPO DE PANTALLA - CREACIÓN/EDICIÓN DE UNA ACTIVIDAD (ELABORACION PROPIA)	42
ILUSTRACIÓN 12. PROTOTIPO DE PANTALLA - DIGITACIÓN DE NOTAS PARA UNA ACTIVIDAD (ELABORACIÓN PROPIA)	43
ILUSTRACIÓN 13. PROTOTIPO DE PANTALLA - CONSULTA DE CALIFICACIONES (ELABORACIÓN PROPIA)	43
ILUSTRACIÓN 14. PROTOTIPO DE PANTALLA - CONSULTA DE ESTUDIANTES A CARGO DE UN DOCENTE (ELABORACIÓN PROPIA)	44
ILUSTRACIÓN 15. PROTOTIPO DE PANTALLA - CONSULTA DE OBSERVACIONES DE UN ESTUDIANTE (ELABORACIÓN PROPIA)	44
ILUSTRACIÓN 16. PORTAL ADMINISTRADOR (ELABORACIÓN PROPIA)	45
ILUSTRACIÓN 17. PORTAL DOCENTE (ELABORACIÓN PROPIA)	46
ILUSTRACIÓN 18. PORTAL ESTUDIANTE (ELABORACIÓN PROPIA)	47
ILUSTRACIÓN 19. PORTAL ACUDIENTES (ELABORACIÓN PROPIA)	48
ILUSTRACIÓN 20. INICIO DE SESIÓN (ELABORACIÓN PROPIA)	48

ÍNDICE DE TABLAS

TABLA 1. CRONOGRAMA DE ACTIVIDADES DEL PROYECTO (ELABORACIÓN PROPIA)	20
TABLA 2. LISTA DE REQUERIMIENTOS (ELABORACIÓN PROPIA).	21
TABLA 3. CASO DE USO 1 (ELABORACIÓN PROPIA).....	23
TABLA 4. CASO DE USO 2 (ELABORACIÓN PROPIA).....	24
TABLA 5. CASO DE USO 3 (ELABORACIÓN PROPIA)	24
TABLA 6. CASO DE USO 4 (ELABORACIÓN PROPIA).....	25
TABLA 7. CASO DE USO 5 (ELABORACIÓN PROPIA).....	26
TABLA 8. CASO DE USO 6 (ELABORACIÓN PROPIA).....	27
TABLA 9. CASO DE USO 7 (ELABORACIÓN PROPIA).....	28
TABLA 10. CASO DE USO 9 (ELABORACIÓN PROPIA).....	28
TABLA 11. CASO DE USO 10 (ELABORACIÓN PROPIA).....	29
TABLA 12. CASO DE USO 11 (ELABORACIÓN PROPIA).....	30
TABLA 13. CASO DE USO 12 (ELABORACIÓN PROPIA).....	31
TABLA 14. CASO DE USO 13 (ELABORACIÓN PROPIA).....	32
TABLA 15. CASO DE USO 14 (ELABORACIÓN PROPIA).....	32
TABLA 16. CASO DE USO 1 (ELABORACIÓN PROPIA).....	33
TABLA 17. CASO DE USO 16 (ELABORACIÓN PROPIA).....	34
TABLA 18. CASO DE USO 17 (ELABORACIÓN PROPIA).....	35
TABLA 19. CASO DE USO 18 (ELABORACIÓN PROPIA).....	35
TABLA 20. CASO DE USO 19 (ELABORACIÓN PROPIA).....	36

Anexos

Anexo: Instructivo de puesta en marcha del sistema

1 INTRODUCCIÓN

La Institución Educativa Agustín Nieto Caballero es una institución educativa que se ha propuesto mejorar y ser una institución de excelente calidad que se basa en principios éticos y sociales para la formación de futuros ciudadanos de bien que aporte al crecimiento y desarrollo de la sociedad. Es por ello que para garantizar este objetivo es indispensable afrontar las diversas problemáticas que se presenten a lo largo del proceso.

La deficiencia académica que se manifiesta en algunos estudiantes de la institución, hace presente la necesidad de que padres de familia se mantengan al tanto de la situación académica de sus hijos en el transcurso del año lectivo que estén cursando, de ésta manera, se ve oportuno la implementación de un sistema de gestión de notas que permita esto último.

2 DEFINICIÓN DEL PROBLEMA

2.1 DESCRIPCIÓN O PLANTEAMIENTO DEL PROBLEMA

La Institución Educativa Agustín Nieto Caballero, al querer mejorar como institución ve la necesidad de solventar el problema de la deficiencia académica de algunos de sus estudiantes activos en la institución, los padres de familia no tienen conocimiento oportuno y habitual del rendimiento académico de su(s) hijo(s), sino hasta el día de la entrega del boletín de un respectivo periodo académico, haciendo casi imposible que se pueda adoptar medidas respectivas en el momento oportuno y de tal manera lograr mitigar el presente problema, es entonces donde se contempla la posibilidad de brindar un sistema web íntegro, seguro, flexible y de fácil uso que brinde eficazmente un servicio de gestión de notas, y brinde acceso completo a los padres de familia sobre los resultados académicos que sus hijo(s) que vayan obteniendo durante la ejecución del año escolar.

3 JUSTIFICACIÓN

El colegio Agustín Nieto Caballero en sus 38 años de operación como institución educativa, ha visto la necesidad de gestionar parte de sus procesos académicos que son las calificaciones y observaciones que se realizan en la ejecución de los cursos, haciendo uso de la tecnología moderna, y es por eso que demanda el desarrollo de un software de naturaleza web que brinde la creación, edición, organización y disponibilidad de la información relacionada a la ejecución de sus cursos.

4 OBJETIVOS

4.1 OBJETIVO GENERAL

Estructurar y diseñar un sistema de gestión de notas, para la institución educativa Agustín Nieto Caballero.

4.2 Objetivos específicos

- Recopilar y analizar información, organizarla de forma detallada y así delimitar el alcance y los requerimientos del sistema.
- Diseñar y desarrollar vistas de uso intuitivo con base al antiguo modelo de notas y observaciones.
- Diseñar el modelo de datos.
- Realizar las pruebas respectivas que propone la ingeniería del software para la detección de defectos.
- Investigar y definir framework de desarrollo web que se adapte y satisfaga las necesidades presentes.

5 MARCO REFERENCIAL

5.1 MARCO CONTEXTUAL

5.1.1 RESEÑA HISTÓRICA DE LA INSTITUCIÓN.

El instituto Agustín Nieto Caballero, ubicado en la vereda Frailes, municipio de Dosquebradas, fue creado por decreto 103 del 28 de febrero de 1977 emanado de la secretaría de educación del departamento de Risaralda, cuando el gobernador era el señor Gonzalo Vallejo y la secretaría de educación la señora Consuelo de Fatsman. Colegio rural mixto con un grupo de primero de bachillerato para el cual fueron matriculados 25 alumnos y fue nombrado como rector al señor EMILIO SERNA, como secretaria-tesorera la señora María Nancy Cano y profesor a Héctor Hernán Rendón. Se iniciaron labores el 23 de marzo de 1977, en un aula construida especialmente para este grupo con el comité de cafeteros. El 2 de noviembre llega la visita de supervisión para la aprobación de este grado. Se consiguió un puntaje de 75.12%. En 1978 se creó el grado segundo y se nombraron dos profesores. En 1979 se crea el grado tercero y llegaron tres nuevos profesores. En 1980 se crea el grado cuarto. En 1983 permuta el señor EMILIO SERNA con la señora CARMEN NUBIA OSPINA de Belén de Umbría. Otro grado primero se formó en febrero 17 de 1983 y se consiguieron dos plazas docentes en comisión. En octubre de 1989 se empezaron los trabajos de construcción de la nueva planta física en la escuela de Frailes para pasar el colegio a este lugar. El traslado del colegio a las nuevas instalaciones (ubicación actual) se realizó el 25 de julio de 1986. En abril 28,29 y 30 de 1987 los supervisores de educación realizan una visita para aprobación de estudios, obteniendo el colegio un puntaje de 90.34%.

En el año 1991 se abre espacio para doble jornada creándose la de la tarde. En 1992 surge la jornada nocturna. En el año 2002, mediante resolución 2527 de noviembre 22 y atendiendo a las orientaciones del ministerio de educación nacional, se crea la institución educativa AGUSTÍN NIETO CABALLERO, albergando las escuelas Santiago Londoño y Frailes.

La primera promoción de bachilleres de la jornada de la mañana fue en el año de 1991 y la de la tarde en el año de 1994.

En la actualidad la institución cuenta con una muy buena estructura física, (aula máxima, coliseo, sala de profesores, oficinas de: rectoría, secretaría y tesorería, biblioteca, cafetería, recepción, aula de tecnología, sala de sistemas, cuenta con una banda músico-marcial.

La institución opera bajo los lineamientos de la constitución política de Colombia, la ley general de educación y decretos que regulan. Así como el código del menor entre otros, para construir y formar personas en RESPETO, RESPONSABILIDAD y TOLERANCIA.

5.2 MARCO CONCEPTUAL

Sitio web:

Nacimiento de la web

Tim Berners-Lee un científico británico en CERN, que inventó el World Wide Web en 1989. La web fue originalmente concebida y desarrollada con fin de satisfacer la necesidad de compartir información entre científicos de universidades e institutos alrededor del mundo de forma automatizada.[1]

w3c (world wide web consortium), que es una comunidad internacional que se encarga de desarrollar estándares web, define sitio web como:

Una colección de páginas web interconectadas, incluida una página de host, que residen en la misma ubicación de red. Por "interconectado" se entiende que se puede acceder a cualquiera de las páginas web integrantes de un sitio web siguiendo una secuencia de referencias que comienza en la página de host del sitio; abarca cero, una o más páginas web ubicadas en el mismo sitio; y terminando en la página web en cuestión.[2]

Bases de datos:

Una base de datos es una colección de información que está organizada para que se pueda acceder, administrar y actualizar fácilmente.

Los datos están organizados en filas, columnas y tablas, y están indexados para facilitar la búsqueda de información relevante. Los datos se actualizan, expanden y eliminan a medida que se agrega nueva información. Las bases de datos procesan cargas de trabajo para crearse y actualizarse, consultar los datos que contienen y ejecutar aplicaciones en su contra.

Las bases de datos informáticas suelen contener agregaciones de registros o archivos de datos, como transacciones de ventas, catálogos e inventarios de productos y perfiles de clientes.

Por lo general, un administrador de bases de datos proporciona a los usuarios la capacidad de controlar el acceso de lectura / escritura, especificar la generación de informes y analizar el uso. Algunas bases de datos ofrecen conformidad ACID (atomicidad, coherencia, aislamiento y durabilidad) para garantizar que los datos sean coherentes y que las transacciones estén completas.[3]

Base de datos relacional

Es una base de datos tabular en la que los datos se definen de manera que se pueden reorganizar y acceder de diferentes maneras.

Las bases de datos relacionales se componen de un conjunto de tablas con datos que se ajustan a una categoría predefinida. Cada tabla tiene al menos una categoría de datos en una columna, y cada fila tiene una determinada instancia de datos para las categorías que se definen en las columnas.[3]

MySQL

MySQL es un sistema de gestión de bases de datos relacionales de código abierto (RDBMS) basado en Structured Query Language (SQL).

MySQL se ejecuta en prácticamente todas las plataformas, incluidos Linux, UNIX y Windows. Aunque se puede utilizar en una amplia gama de aplicaciones, MySQL suele asociarse con aplicaciones basadas en la web y publicaciones en línea, y es un componente importante de una pila open source enterprise llamada LAMP.

LAMP es una plataforma de desarrollo web que utiliza Linux como sistema operativo, Apache como el servidor web, MySQL como el sistema de gestión de bases de datos relacionales y PHP como el lenguaje de scripting orientado a objetos. (A veces se usa Perl o Python en lugar de PHP).

MySQL, que fue originalmente concebido por la compañía sueca MySQL AB, fue adquirida por Sun Microsystems en 2008 y luego por Oracle cuando compró Sun en 2010. Los desarrolladores aún pueden usar MySQL bajo la Licencia Pública General de GNU (GPL), pero las empresas deben obtener una licencia comercial de Oracle.[4]

Lenguaje de programación:

PHP:

PHP tal y como se conoce hoy en día es en realidad el sucesor de un producto llamado PHP/FI. Creado en 1994 por Rasmus Lerdorf, la primera encarnación de PHP era un conjunto simple de ficheros binarios Common Gateway Interface (CGI) escritos en el lenguaje de programación C. Originalmente utilizado para rastrear visitas de su currículum online, llamó al conjunto de scripts "Personal Home Page Tools", más frecuentemente referenciado como "PHP Tools". Con el paso del tiempo se quiso más funcionalidad, y Rasmus reescribió PHP Tools, produciendo una implementación más grande y rica. Este nuevo modelo fue capaz de interactuar con bases de datos, y mucho más, proporcionando un entorno de trabajo sobre cuyos usuarios podían desarrollar aplicaciones web dinámicas sencillas tales como libros de visitas.[5]

¿Qué es php?

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

Ejemplo ilustrativo

```
<!DOCTYPE HTML>

<html>

    <head>

        <title>Ejemplo</title>

    </head>

    <body>

        <?php    echo "¡Hola, soy un script de PHP!";    ?>

    </body>

</html>
```

En lugar de usar muchos comandos para mostrar HTML (como en C o en Perl), las páginas de PHP contienen HTML con código incrustado que hace "algo" (en este caso, mostrar "¡Hola, soy un script de PHP!"). El código de PHP está encerrado entre las etiquetas especiales de comienzo y final <?php y ?> que permiten entrar y salir del "modo PHP".

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo de la manga.[6]

HTML:

HTML es un lenguaje artificial que los ordenadores son capaces de interpretar y diseñado para que los programadores redacten instrucciones que los navegadores ejecutan para originar la página web.

Las siglas HTML significan Hyper Text Markup Language (lenguaje de marcas de hipertexto). El hipertexto en una computadora es texto que posee referencias, hipervínculos, links o enlaces a otro texto. Este concepto surgió de la necesidad de los científicos de procesar la información mecánicamente entre ordenadores conectados.[7]

Servidor web:

El servidor web es un programa que está esperando permanentemente las solicitudes de conexión mediante el protocolo HTTP por parte de los clientes web. La parte servidor de las aplicaciones web está formada por páginas estáticas que siempre muestran el mismo contenido y por programas o scripts que son ejecutados por el servidor web cuando el navegador del cliente solicita algunas páginas. La salida de este script suele ser una página HTML estándar que se envía al navegador del cliente. Tradicionalmente este programa o script que es ejecutado por el servidor web se basa en la tecnología Common Gateway Interface (CGI).

Framework:

En general, con el término framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.[9]

Laravel:

Laravel es un framework de desarrollo de software de naturaleza web que facilita a los programadores el desarrollo al proponer código reutilizable acelerando así el desarrollo. Algunas de las características del framework se listan a continuación:

- **Modularidad:** Laravel se construyó sobre más de 20 bibliotecas diferentes y se divide en módulos individuales. Completamente integrados con el administrador de dependencias del compositor, estos componentes se pueden actualizar con facilidad.
- **Testabilidad:** construido desde cero para facilitar las pruebas, laravel se envía con varios helpers que le permiten visitar las rutas de su prueba, rastrear el HTML resultante, asegurarse de que se invoquen métodos en ciertas clases e incluso hacerse pasar por usuarios autenticados para asegurarse de que el código correcto se ejecuta en el momento adecuado.
- **Routing:** Laravel le da mucha flexibilidad cuando define las rutas de su aplicación. Por ejemplo, puede vincular manualmente una función anónima simple a una ruta con un verbo HTTP, como GET, POST, PUT o DELETE. Esta función está inspirada en microarmas, como Sinatra (Ruby) y Siles (PHP). [10]
- **Query builder y ORM:** Laravel se envía con un generador de consultas fluido, que le permite emitir consultas de bases de datos con una sintaxis PHP, donde simplemente encadena los métodos en lugar de escribir SQL. Además de esto, le proporciona un Eloquent, que es similar a lo que encontrará en Ruby on Rails, para ayudarlo a definir modelos interconectados. Tanto el generador de consultas como el ORM son compatibles con diferentes bases de datos, como PostgreSQL, SQLite, MySQL y SQL Server.[10]

6 METODOLOGÍA

La metodología aplicada al proyecto es la metodología ágil XP (Extreme Programming), ya que nos la que más se adapta a nuestra necesidades y permitirá desarrollar este proyecto de forma rápida, agilizando la productividad del mismo.

Adicional a lo anterior, se pretende con esta metodología, aplicar el conjunto de buenas prácticas y principios básicos que ofrece, con el fin de desarrollar un producto de excelente calidad, que brinde la satisfacción del cliente. Algunos de los principios que se consideran para el desarrollo del proyecto son:

- **Proceso de Planificación:** Es significativo elaborar planes acorde a las necesidades del cliente. Estos planes pueden hacerse diariamente, mensualmente o como la situación lo amerite.
- **Cliente en el Sitio:** En esta metodología es primordial la comunicación, que para este aspecto en particular es la definición claramente de los requerimientos, determinando cuales son las prioridades.
- **Diseño Simple:** Se desea aplicar a cada uno de los módulos a desarrollar, un diseño fácil de hacer pero que a su vez sea funcional y cumpla con los requerimientos que se solicitan.
- **Refactorización:** Los programadores de esta metodología están en la capacidad de ver constantemente el diseño del sistema y modificar el código de lo que sea necesario.

Primera fase: Planificación del proyecto

En esta primera fase se hizo, primero una recopilación de la información de la institución, con esta información se estableció: los requerimientos del sistema, se realizó una especificación de estos a través de casos de uso y se realizaron los respectivos diagramas.

Segunda fase: Diseño

En esta fase se realizó el diseño de la base de datos, haciendo uso del modelo relacional y los prototipos de pantalla.

Tercera fase: Codificación

En esta tercera fase se realizó la codificación de los diferentes módulos que componen el sistema de acuerdo a los requerimientos ya definidos.

Cuarta fase: Pruebas

En esta última fase se realizó las pruebas a cada módulo

Cronograma																
Actividades	Semanas															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Recopilación de información																
Consolidar información																
Levantamiento de requerimientos																
Análisis de requerimientos																
Realización de diagrama de casos de uso																
Especificación de requerimientos(Casos de uso)																
Diseño de base de datos																
Realización de documento de trabajo de grado																
Diseño de prototipos de pantalla																
Codificación del prototipo web																
Pruebas del prototipo																

Tabla 1. Cronograma de actividades del proyecto (Elaboración propia)

7 DESARROLLO DEL PROYECTO

7.1 TOMA DE REQUERIMIENTOS

Para el levantamiento de los requerimientos se utilizó la observación directa junto con la entrevista con los docentes y personas involucradas, en las instalaciones de la institución.

7.1.1 LISTA DE REQUERIMIENTOS

ID	Descripción
REQ-01	Permitir al sistema ingresar al sistema con el rol de administrador
REQ-02	Permitir ingresar al sistema con con el rol de docente
REQ-03	Permitir ingresar al sistema con el rol estudiante
REQ-04	Permitir ingresar al sistema con el rol de acudiente
REQ-05	Gestionar usuarios(creación, edición, consulta y eliminación)
REQ-06	Gestionar jornadas(creación, edición, consulta y eliminación)
REQ-07	Gestionar grados(creación, edición, consulta y eliminación)
REQ-08	Gestionar grupos(creación, edición, consulta y eliminación)
REQ-09	Gestionar cursos(creación, edición, consulta y eliminación)
REQ-10	Gestionar periodos(creación, edición, consulta y eliminación)
REQ-11	Gestionar actividades(creación, edición, consulta y eliminación)
REQ-12	Gestionar calificaciones(creación, edición, consulta y eliminación)

Tabla 2. Lista de requerimientos (Elaboración propia).

7.2 CASOS DE USO

7.2.1 ROLES DEL SISTEMA

Administrador: Es el usuario que realiza la gestión de jornadas, grados, grupos, cursos, períodos y administra las cuentas de otros usuarios del sistema.

Docente: Usuario que ingresa al panel de docentes donde puede gestionar actividades, calificaciones y observaciones a cada estudiante de los cursos que tiene asignados

Estudiante: Usuario que puede ingresar al panel de estudiantes y consultar las calificaciones correspondientes a cada una de las actividades programas en los cursos del grupo y grado donde se encuentra matriculado

Acudiente: Usuario que puede ingresar al sistema, realizar consultas de las calificaciones de las calificaciones correspondientes a los usuarios tipo estudiante que estén asociados a él.

7.2.2 DESCRIPCIÓN DE CASOS DE USO

7.2.2.1 ADMINISTRADOR

UC-01	Gestionar Jornada
Actores	Administrador
Tipo	Esencial
Propósito	Le permite al administrador crear, editar y eliminar jornadas de algún año respectivo.
Resumen	El administrador podrá acceder al sistema y gestionar las jornadas que se tendrán en cuenta para algún año seleccionado.
Precondición	Iniciar sesión

Curso normal de eventos	Paso	Acción
	1	El administrador crea, edita o elimina jornadas para algún año seleccionado.
	2	El sistema registra la(s) acción(es) realizada(s)
Poscondición		
Curso alternativo de eventos o excepciones	Paso	Acción
	1	El administrador crea más de dos jornadas
	2	El sistema despliega un mensaje de error al respecto.

Tabla 3. Caso de uso 1 (Elaboración propia)

UC-02	Gestionar periodo académico	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador crear, editar y eliminar periodos académicos respectivos a un año..	
Resumen	El administrador podrá acceder al sistema y gestionar los periodos académicos de un año respectivo.	
Precondición	Iniciar sesión	
Curso normal de eventos	Paso	Acción
	1	El administrador crea, edita o elimina periodos académicos de un año seleccionado.
	2	El sistema registra la(s) acción(es) realizada(s)
Poscondición		
Curso alternativo de eventos o excepciones	Paso	Acción
	1	El administrador selecciona una fecha de inicio que se cruza con un periodo anterior.

	2	El sistema no permite crear el período y notifica el error de cruce de fechas.
--	----------	--

Tabla 4. Caso de uso 2 (Elaboración propia)

UC-03	Gestionar Usuarios	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador crear, editar, consultar y eliminar cuentas de usuario de docentes, estudiantes y acudientes.	
Resumen	El administrador podrá acceder al sistema y gestionar cuentas de usuarios tipo docentes, estudiantes y acudientes	
Precondición	Iniciar sesión	
Curso normal de eventos	Paso	Acción
	1	El administrador selecciona la opción “administrar cuentas”
	2	El sistema despliega los tres tipos de cuentas que puede administrar: Docentes, Estudiantes y Acudientes
	3	El administrador selecciona un tipo de cuenta: Docentes, Estudiantes y Acudientes
	4	El sistema muestra los usuarios ya creados y las opciones: Crear usuario Eliminar Usuario Editar Usuario Consultar Usuario
	5	El administrador selecciona una de las opciones y se activa el caso de uso respectivo
Poscondición		
Curso alternativo de eventos o excepciones	Paso	Acción
	1	Si no se han creado usuarios el sistema solo mostrará la opciones: Crear usuario Eliminar Usuario Editar Usuario Consultar Usuario

Tabla 5. Caso de uso 3 (Elaboración propia)

UC-04	Crear Usuario	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador crear cuenta de usuario tipo docente, estudiante o acudiente.	
Resumen	El administrador podrá crear una cuenta de usuario tipo docente, estudiante o acudiente y suministrar los datos al respecto.	
Precondición	Iniciar sesión, Gestionar usuarios (docente,estudiante,acudiente)	
Curso normal de eventos	Paso	Acción
	1	El administrador selecciona la opción "Nuevo profesor/estudiante/acudiente"
	2	El sistema despliega la vista con los campos que almacenará los datos.
	3	El administrador digita los datos respectivos.
	4	El administrador registra.
Poscondición		
Curso alternativo de eventos o excepciones	Paso	Acción

Tabla 6. Caso de uso 4 (Elaboración propia)

UC-05	Editar Usuario	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador editar cuenta de usuario tipo docente, estudiante o acudiente.	
Resumen	El administrador podrá editar una cuenta de usuario tipo docente, estudiante o acudiente para realizar actualizaciones de los datos al respecto.	

Precondición	Iniciar sesión, Gestionar usuarios (docente,estudiante,acudiente)	
Curso normal de eventos	Paso	Acción
	1	El administrador da clic en la opción "editar" al frente de la cuenta tipo profesor/estudiante/acudiente"
	2	El sistema despliega la vista con los campos y los datos suministrados en una sesión anterior.
	3	El administrador cambia datos respectivos y selecciona la opción "registrar"
	4	El sistema valida los campos editados y muestra un mensaje de que se actualizo correctamente
Poscondición		
Curso alternativo de eventos o excepciones	Paso	Acción
	3	El administrador selecciona la opción cancelar
	4	El sistema regresa a la vista de gestión de cuentas

Tabla 7. Caso de uso 5 (Elaboración propia)

UC-06	Eliminar Usuario	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador eliminar cuenta de usuario tipo docente, estudiante o acudiente.	
Resumen	El administrador podrá eliminar una cuenta de usuario tipo docente, estudiante o acudiente.	
Precondición	Iniciar sesión, Gestionar usuarios (docente,estudiante,acudiente)	
Curso normal de eventos	Paso	Acción
	1	El administrador da clic en la opción "eliminar" al frente de la cuenta tipo profesor/estudiante/acudiente"
	2	El sistema solicita suministrar la contraseña del administrador.

	3	El administrador suministra la contraseña y acepta.
	4	El sistema elimina la cuenta de la base de datos y muestra mensaje de que se ha eliminado satisfactoriamente
Poscondición		
Curso alternativo de eventos o excepciones	Paso	Acción
	3	El administrador suministra una contraseña incorrecta
	4	El sistema muestra mensaje de error de contraseña no válida y solicita de nuevo la contraseña del administrador

Tabla 8. Caso de uso 6 (Elaboración propia)

UC-07	Gestionar grados	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador crear, editar, consultar y eliminar grados	
Resumen	El administrador podrá acceder al sistema y gestionar los grados	
Precondición	Iniciar sesión	
Curso normal de eventos	Paso	Acción
	1	El administrador selecciona la opción "gestionar grados"
	2	El sistema muestra los grados ya creados y las opciones: Crear grado Eliminar grado Editar grado
	3	El administrador selecciona una de las opciones y se activa el caso de uso respectivo
Poscondición		
Curso alternativo de eventos excepciones	Paso	Acción
	2	Si no se han creado usuarios el sistema solo mostrará la opciones:

		Crear grado Eliminar grado Editar grado
--	--	--

Tabla 9. Caso de uso 7 (Elaboración propia)

UC-09	Gestionar asignaturas	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador crear, editar, consultar y eliminar asignaturas	
Resumen	El administrador podrá acceder al sistema y gestionar las asignaturas	
Precondición	Iniciar sesión	
Curso normal de eventos	Paso	Acción
	1	El administrador selecciona la opción "gestionar asignatura"
	2	El sistema muestra los grados ya creados y las opciones: Crear asignatura Eliminar asignatura Editar asignatura
	3	El administrador selecciona una de las opciones y se activa el caso de uso respectivo
Poscondición		
Curso alterno de eventos	Paso	Acción
	2	Si no se han creado usuarios el sistema solo mostrará la opciones: Crear asignatura Eliminar asignatura Editar asignatura

Tabla 10. Caso de uso 9 (Elaboración propia)

UC-10	Gestionar cursos	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador crear, editar, consultar y eliminar cursos	
Resumen	El administrador podrá acceder al sistema y gestionar los cursos	
Precondición	Iniciar sesión	
Curso normal de eventos	Paso	Acción
	1	El administrador selecciona la opción "gestionar asignatura"
	2	El sistema muestra los grados ya creados y las opciones: Crear curso Eliminar curso Editar curso
	3	El administrador selecciona una de las opciones y se activa el caso de uso respectivo
Poscondición		
Curso alterno de eventos	Paso	Acción
	2	Si no se han creado usuarios el sistema solo mostrará las opciones: Crear curso Eliminar curso Editar curso

Tabla 11. Caso de uso 10 (Elaboración propia)

UC-11	Registrar cursos	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador registrar/asociar un curso a un grupo específico	

Resumen	El administrador podrá acceder al sistema y registrar los distintos cursos que se dictan	
Precondición	Iniciar sesión, Gestionar grupos	
Curso normal de eventos	Paso	Acción
	1	El administrador selecciona la opción registrar grupo
	2	El sistema muestra los campos de grupo y curso
	3	El administrador selecciona el grupo, el curso que desea registrar/asociar a este y selecciona la opción "Guardar"
	4	
Poscondición		
Curso alterno de eventos	Paso	Acción
	1	Si no se han creado usuarios el sistema solo mostrará las opciones: Crear curso Eliminar curso Editar curso

Tabla 12. Caso de uso 11 (Elaboración propia)

UC-12	Registrar estudiante	
Actores	Administrador	
Tipo	Esencial	
Propósito	Le permite al administrador registrar estudiante a un grupo específico.	
Resumen	El administrador asigna los estudiantes que compondrán los grupos respectivos.	
Precondición	Iniciar sesión, Gestionar usuarios	
Curso normal de eventos	Paso	Acción
	1	El administrador selecciona "Estudiantes"

	2	El sistema muestra la opción de creación de usuario.
	3	El administrador crea una cuenta estudiante.
	4	El sistema muestra los campos respectivos para ingresar el nombre, código, correo, teléfono, y grupo al que pertenecerá el estudiante.
	5	El administrador confirma la creación del estudiante.
	6	El sistema registra a un estudiante dentro del grupo seleccionado.
Poscondición		
Curso alternativo de eventos	Paso	Acción
	1	El administrador, intenta crear un estudiante, pero no existen grupos al que asociarlo.
	2	El sistema le indica que debe asociarlo a un grupo.

Tabla 13. Caso de uso 12 (Elaboración propia)

7.2.2.2 DOCENTE

UC-13	Consultar grupo	
Actores	Docente	
Tipo	Esencial	
Propósito	Permitir al profesor consultar cualquier grupo, donde al menos dicte un curso, dándole la posibilidad de que pueda realizar todas las acciones al respecto requeridas.	
Resumen	El profesor podrá consultar los distintos grupos que dicte al menos un curso asociado, acto seguido podrá realizar todas las acciones pertinentes con dicho grupo, como listar estudiantes, actividades que haya agregado, las calificaciones respectivas y las observaciones generales hacia cada estudiante.	
Precondición	Iniciar sesión	
Curso normal de eventos	Paso	Acción
	1	El profesor hace clic en consultar grupo.

	2	El sistema despliega la vista respectiva, que contendrá tres listas desplegables (grado, grupo, curso), también mostrará 3 opciones (listar estudiantes, actividades y observaciones). La vista por defecto será las actividades que tenga el docente con el grupo y curso respectivo.
Poscondición		
Curso alternativo de eventos	Paso	Acción
	1	Si el profesor no tiene al menos un curso para dictar, esta vista mostrará los campos asociados vacíos.

Tabla 14. Caso de uso 13 (Elaboración propia)

UC-14	Listar estudiantes	
Actores	Docente	
Tipo	Esencial	
Propósito	Permitir a los profesores obtener las lista de estudiantes de un curso y grupo específico.	
Resumen	Los profesores podrán acceder a la lista de estudiantes de un curso que dicten para el grupo respectivo, donde tendrá acceso a información como nombre y código.	
Precondición	Iniciar sesión, Consultar grupo	
Curso normal de eventos	Paso	Acción
	1	El profesor hace clic en la opción listar estudiantes
	2	El sistema muestra la lista de estudiantes que pertenecen a un grupo y curso respectivo con la respectiva información.
Poscondición		
Curso alternativo de eventos	Paso	Acción

Tabla 15. Caso de uso 14 (Elaboración propia)

UC-15	Gestionar actividades	
Actores	Docente	
Tipo	Esencial	
Propósito	El profesor podrá acceder a las actividades que haya agregado a un curso de algún grupo respectivo.	
Resumen	El profesor podrá acceder a las actividades que haya agregado a algún curso durante su desarrollo, y poder realizar acciones como borrar, editar, agregar subactividad, delegar porcentaje de peso y digitar la nota.	
Precondición	Iniciar sesión, Consultar grupo	
Curso normal de eventos	Paso	Acción
	1	El profesor hace clic en la opción de listar actividades.
	2	El sistema muestra las actividades así como las distintas acciones que se puedan realizar al respecto.
Poscondición		
Curso alterno de eventos	Paso	Acción

Tabla 16. Caso de uso 1 (Elaboración propia)

UC-16	Gestionar calificaciones	
Actores	Docente	
Tipo	Esencial	
Propósito	El profesor podrá realizar las calificaciones a los estudiantes de algún curso que él dicte.	
Resumen	El profesor podrá realizar las calificaciones de las actividades de su curso que realicen sus estudiantes.	
Precondición	Iniciar sesión, Consultar grupo, Listar actividades.	
	Paso	Acción

Curso normal de eventos	1	El profesor hace clic en la opción editar nota al frente de cada actividad.
	2	El sistema despliega la lista de los estudiantes con un campo de edición de nota al frente de cada estudiante..
Poscondición		
Curso alternativo de eventos	Paso	Acción
	1	El profesor deja el campo de edición de nota vacío de algún estudiante.
	2	El sistema no registra nada sobre la calificación de ese estudiante.

Tabla 17. Caso de uso 16 (Elaboración propia)

UC-17	Gestionar observaciones	
Actores	Docente	
Tipo	Esencial	
Propósito	El profesor podrá realizar las observaciones a los estudiantes de algún curso que él dicte.	
Resumen	El profesor podrá realizar observaciones sobre el rendimiento académico de todos los estudiantes que pertenecen a su curso al final de un periodo académico.	
Precondición	Iniciar sesión, Consultar grupo.	
Curso normal de eventos	Paso	Acción
	1	El profesor hace clic en la opción realizar observaciones.
	2	El sistema despliega la lista de los estudiantes.
	3	El profesor hace clic en un estudiante.
	4	El sistema muestra una ventana con un campo de edición, donde se podrá digitar las observaciones respectivas.
Poscondición		
	Paso	Acción

Curso alternativo de eventos	1	El profesor deja el campo de edición vacío.
	2	El sistema no registra nada.

Tabla 18. Caso de uso 17 (Elaboración propia)

7.2.2.3 ESTUDIANTE - DOCENTE

UC-18	Consultar Calificaciones	
Actores	Estudiante, Acudientes	
Tipo	Esencial	
Propósito	El estudiante o acudiente podrá consultar todas las calificaciones.	
Resumen	El estudiante o acudiente podrá realizar la consulta de sus calificaciones de todos sus cursos del año.	
Precondición	Iniciar sesión.	
Curso normal de eventos	Paso	Acción
	1	El Estudiante o acudiente hace clic en la opción consultar calificaciones.
	2	El sistema despliega las calificaciones del estudiante, segmentadas en sus cursos respectivos.
Poscondición		
Curso alternativo de eventos	Paso	Acción
	1	El estudiante o acudiente consulta calificaciones después de haber aprobado el año.
	2	El sistema no muestra nada.

Tabla 19. Caso de uso 18 (Elaboración propia)

7.2.2.4 Acudiente

UC-19	Consultar observaciones	
Actores	Acudiente	
Tipo	Esencial	
Propósito	El acudiente podrá consultar el estado académico de su hijo(s)/hija(s) mediante observaciones.	
Resumen	El acudiente podrá consultar el estado académico de su hijo(s)/hija(s), según las observaciones que los docentes hayan realizado y registrado.	
Precondición	Iniciar sesión.	
Curso normal de eventos	Paso	Acción
	1	El Acudiente hace clic en la opción consultar observaciones.
	2	El sistema entrega una lista desplegable de los hijo(s)/hija(s) del acudiente.
	3	El acudiente hace clic sobre su hijo/hija
	4	El sistema muestra las observaciones de su hijo/hija.
Poscondición		
Curso alterno de eventos	Paso	Acción
	1	El Acudiente consulta la(s) observación(es) después del tiempo establecido.
	2	El sistema no muestra nada.

Tabla 20. Caso de uso 19 (Elaboración propia)

7.2.3 DIAGRAMAS DE CASOS DE USO

Los diagramas de casos de uso se realizaron con ayuda de la herramienta Star UML, esta tiene una versión gratuita

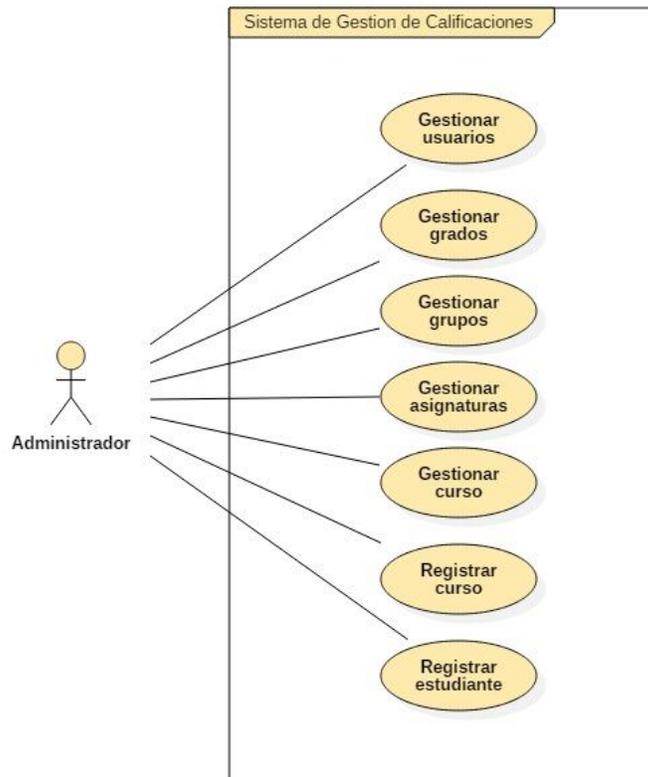


Ilustración 1. Diagrama de casos de uso administrador (Elaboración propia)

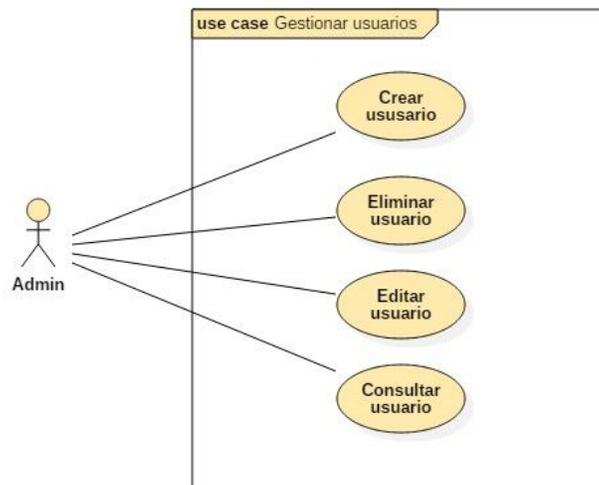


Ilustración 2. Diagrama de casos de uso administrador - Creación de usuarios (Elaboración propia)

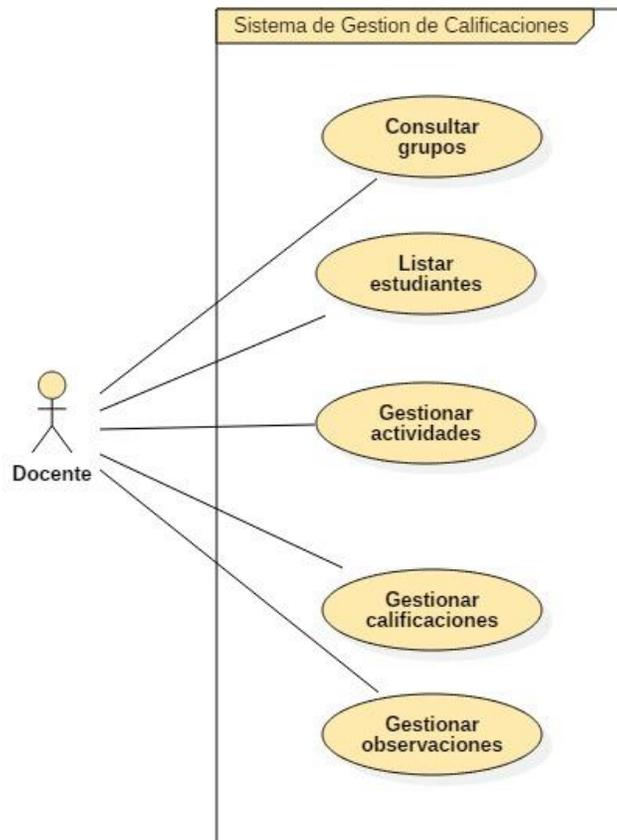


Ilustración 3. Diagrama de casos de uso docente (Elaboración propia)

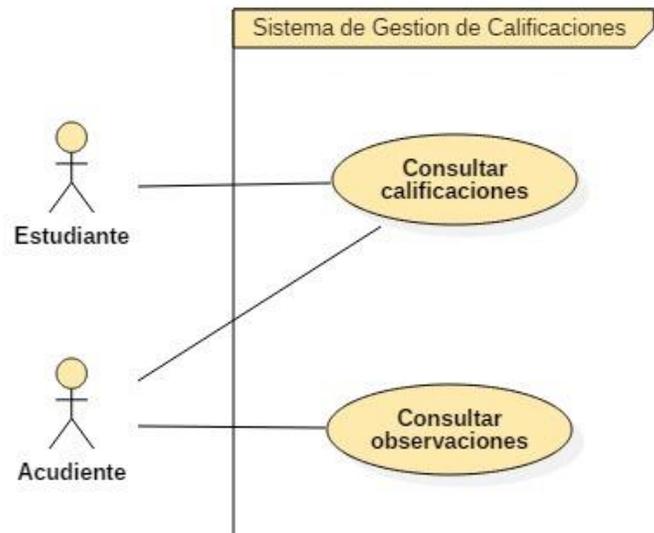


Ilustración 4. Diagrama de casos de uso - Estudiante Acudiente (Elaboración propia)

7.3 DISEÑO DEL MODELO DE DATOS

7.3.1 MODELO DE DATOS DEL SISTEMA

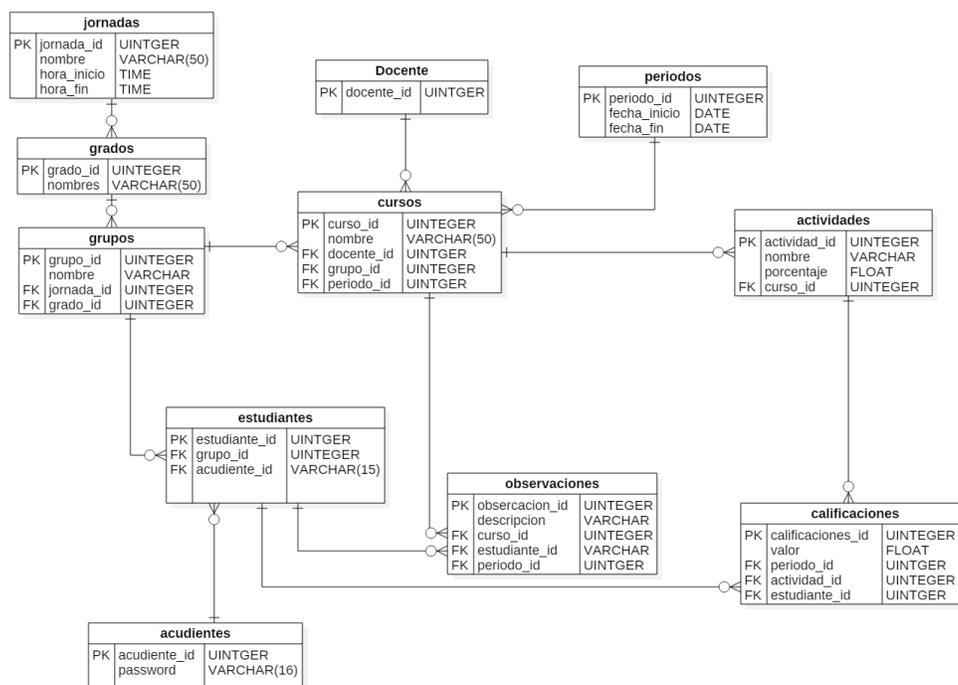


Ilustración 5. Diagrama relacional - Sistema (Elaboración propia)

7.3.2 MODELO DE DATOS PARA EL INICIO DE SESION

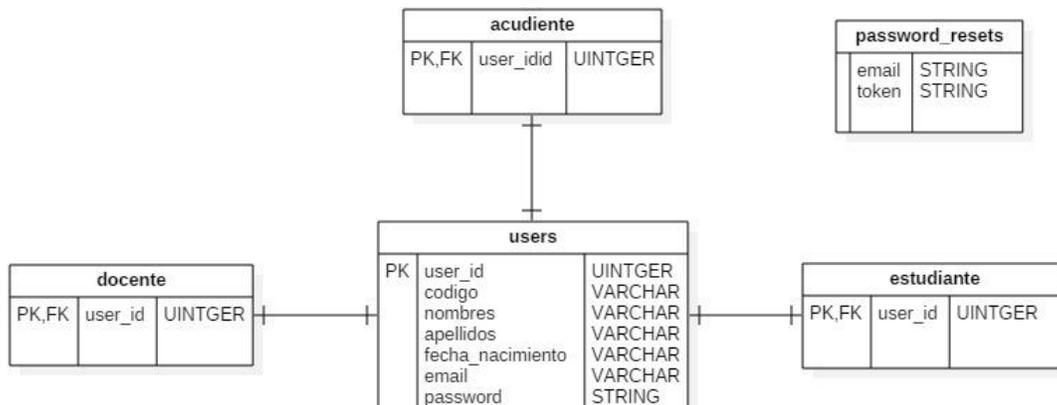


Ilustración 6. Diagrama relacional - Inicio de sesión (Elaboración propia)

7.4 DISEÑO DE PROTOTIPOS DE PANTALLA

A continuación se exponen los prototipos de pantalla más importantes del sistema, hay procedimientos que se omiten por el hecho de ser similares a los expuestos, como es el caso de agregar, consultar, editar y eliminar estudiante o cualquier otro usuario del sistema, estos prototipos se realizaron con la herramienta Balsamiq Mockups 3, esta herramienta es de pago, pero ofrecen una versión de prueba por 30 días.

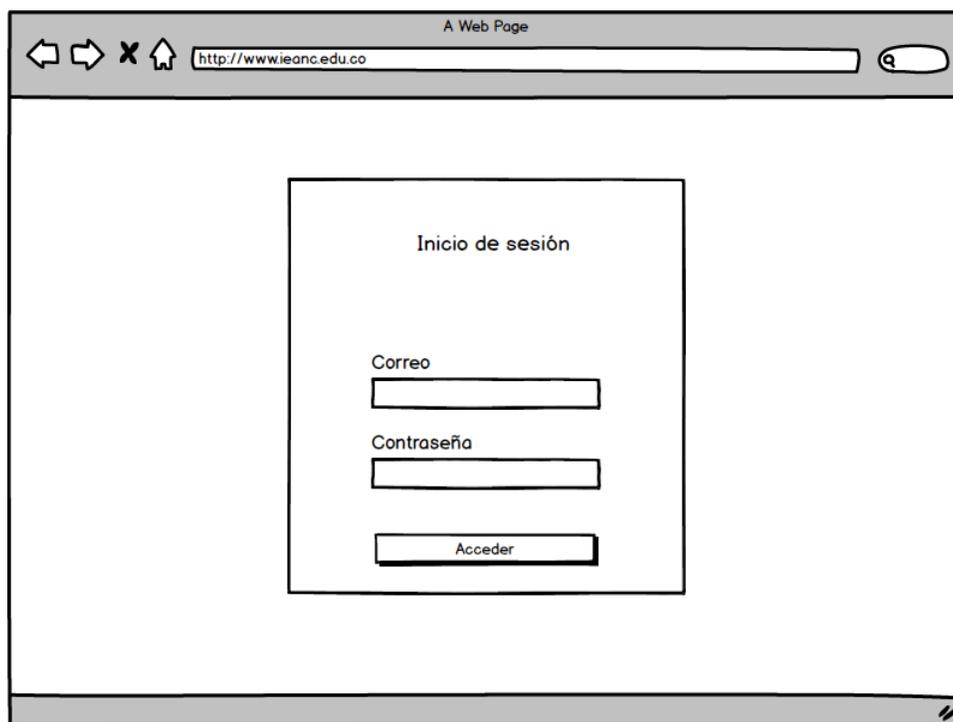


Ilustración 7. Prototipo de pantalla -Inicio de sesión (Elaboración propia)

A Web Page
<http://www.ieanc.edu.co>

LOGO Portal Administrador

Nuevo estudiante

Nombre
 Nombres Apellidos

Cedula de ciudadanía
 Numero cedula

Correo Telefono movil

Genero Fecha de nacimiento

Jornada Grado Grupo

Ilustración 8. Prototipo de pantalla – Creación/Edición de un usuario (Elaboración propia)

A Web Page
<http://www.ieanc.edu.co>

LOGO Portal

Creacion/Edicion de un Grupo

Grado Grupo

Director de grupo

Jornada

Ilustración 9. Pro tipo de pantalla - Creación/Edición de un grupo (Elaboración propia)

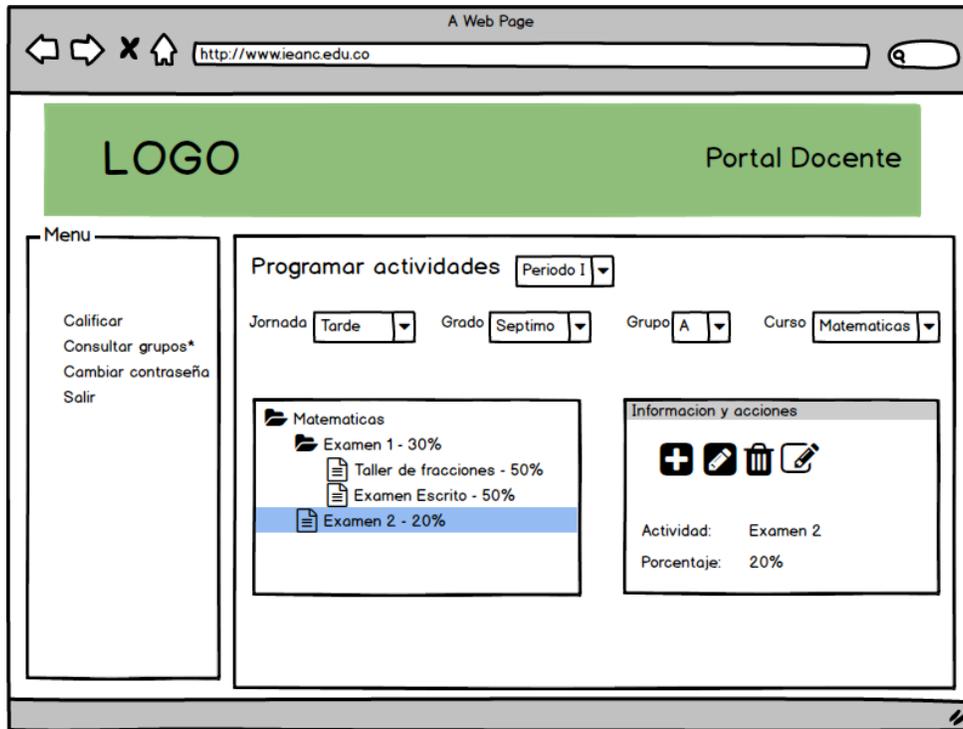


Ilustración 10. Prototipo de pantalla – Gestión de actividades y calificaciones (Elaboración propia)

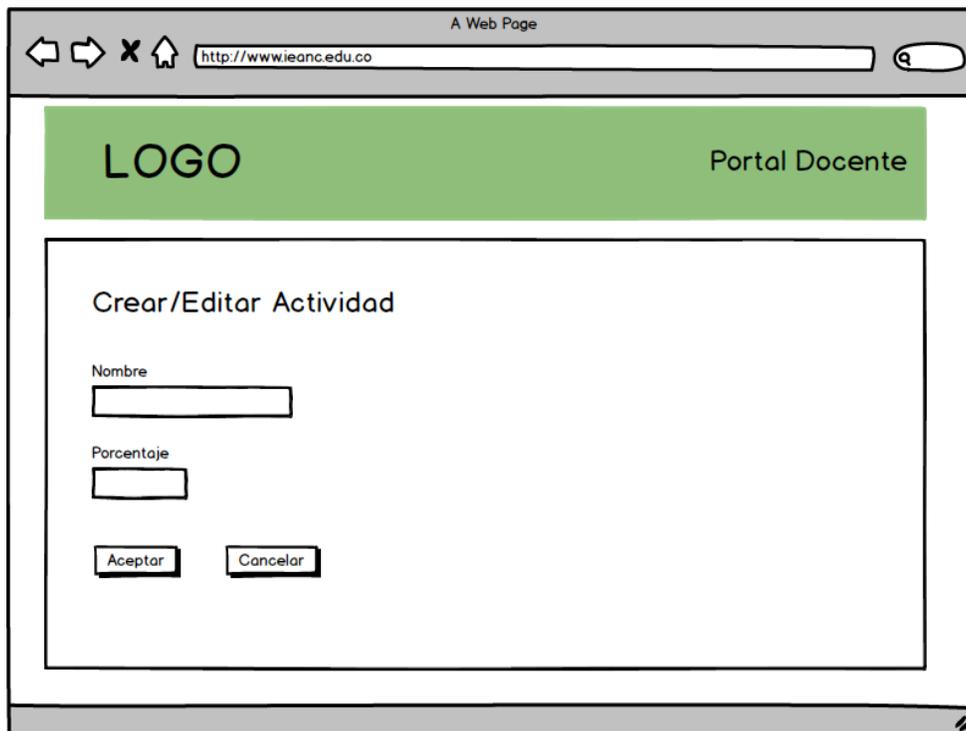


Ilustración 11. Prototipo de pantalla - Creación/Edición de una actividad (Elaboración propia)

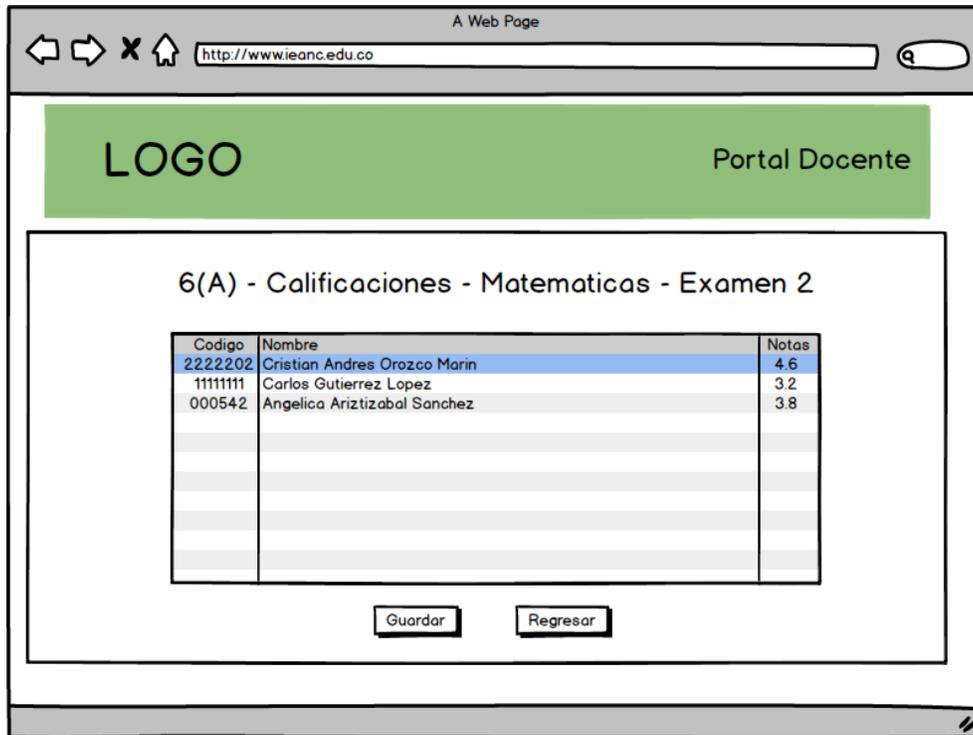


Ilustración 12. Prototipo de pantalla - Digitación de notas para una actividad (Elaboración propia)

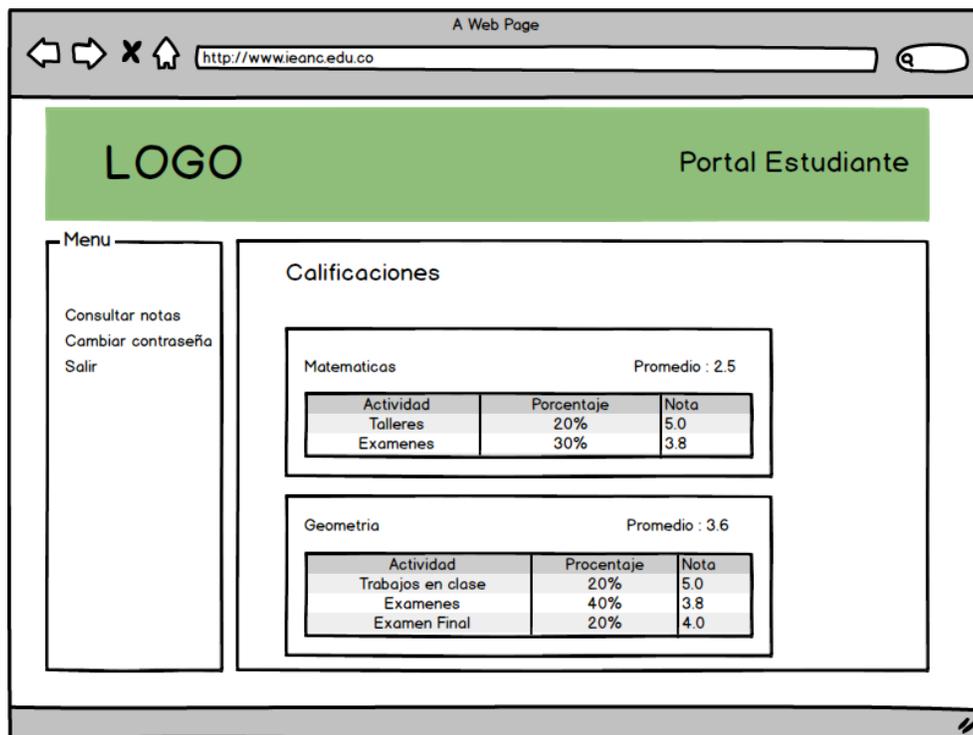


Ilustración 13. Prototipo de pantalla - Consulta de calificaciones (Elaboración propia)

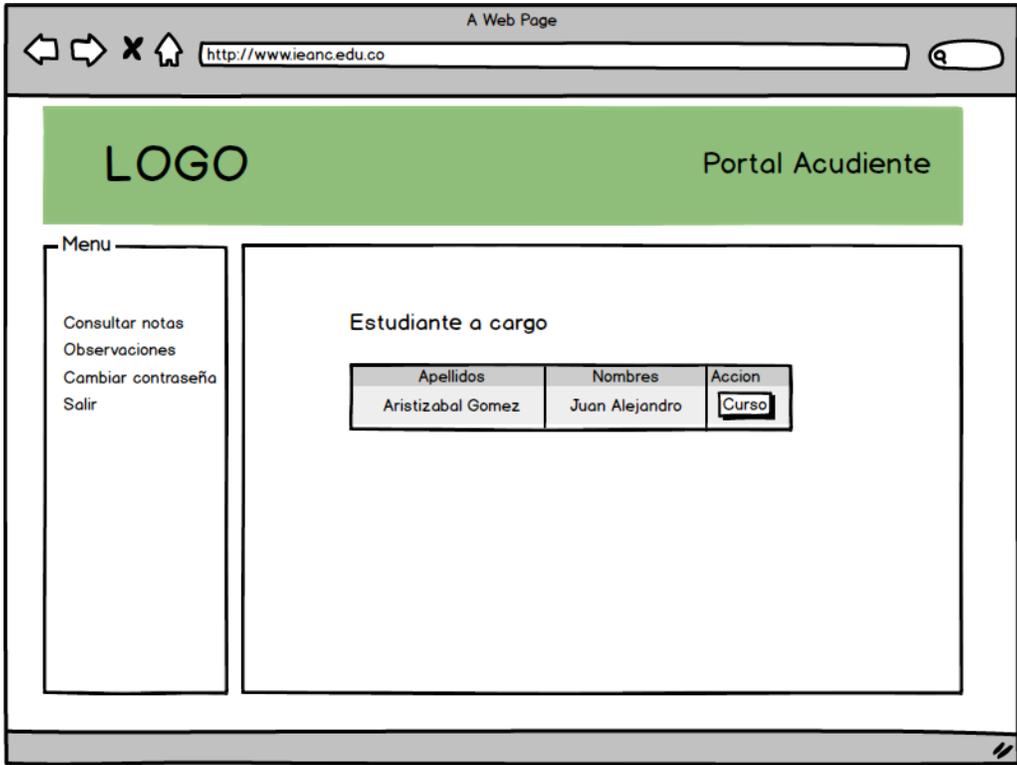


Ilustración 14. Prototipo de pantalla - consulta de estudiantes a cargo de un docente (Elaboración propia)

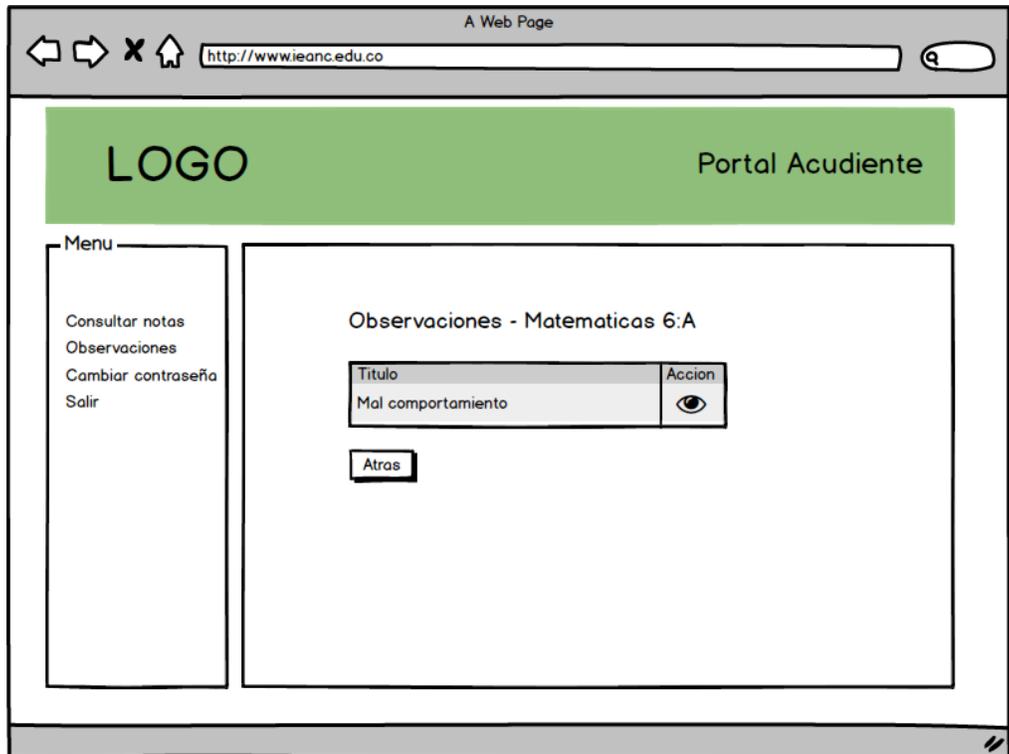


Ilustración 15. Prototipo de pantalla - Consulta de observaciones de un estudiante (Elaboración propia)

7.5 IMPLEMENTACIÓN DEL SISTEMA

El sistema se desarrolló usando el framework Laravel, ya que nos brinda desarrollo ágil con código simple y elegante, para el frontend se usó Jquery y bootstrap y el gestor de base de datos que se utilizó fue MySQL.

7.5.1 MÓDULO DE ADMINISTRADOR

El módulo de administrador está desarrollado para la gestión de la configuración del sistema, desde acá se administran las jornadas, los grados, los cursos y se gestionan los usuarios a los otros usuarios: docentes, estudiantes y acudientes.

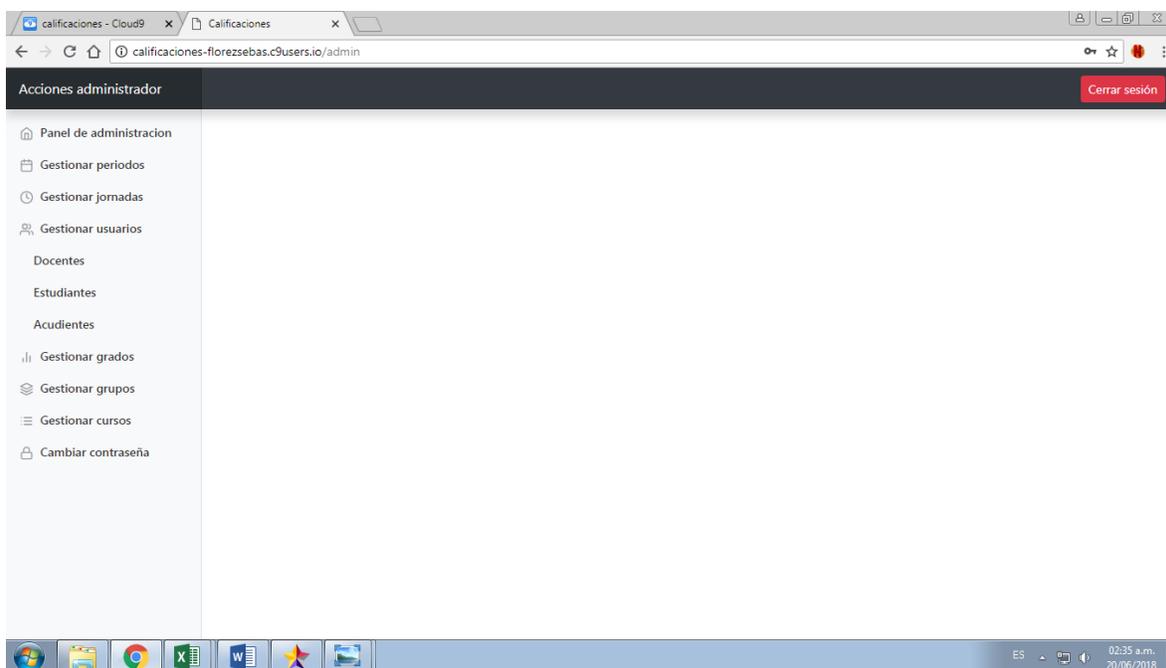


Ilustración 16. Portal administrador (Elaboración propia)

7.5.2 MÓDULO DE DOCENTE

El módulo de docente está desarrollado para la administración de calificaciones, actividades y observaciones de los cursos, esto le permitirá al docente tener un mejor control académico de los estudiantes de la institución.

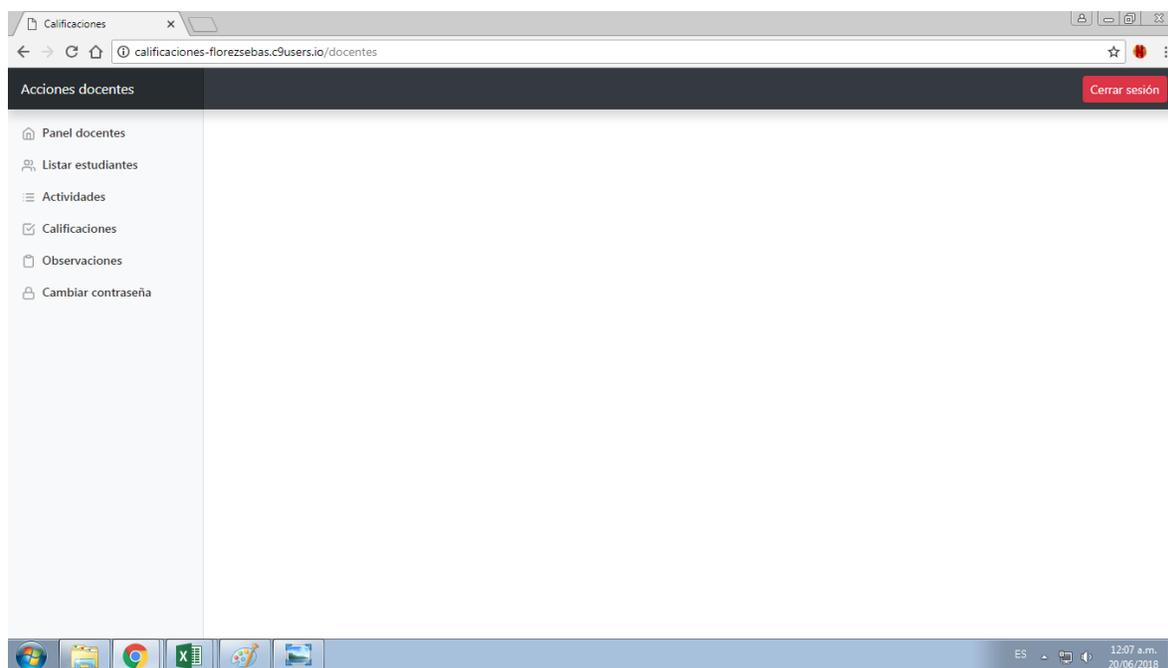


Ilustración 17. Portal docente (Elaboración propia)

7.5.3 MÓDULO DE ESTUDIANTE

El módulo de estudiante está desarrollado para que el estudiante tenga información oportuna de sus calificaciones, esto le permite conocer exactamente la forma en cómo los docentes lo están evaluando.

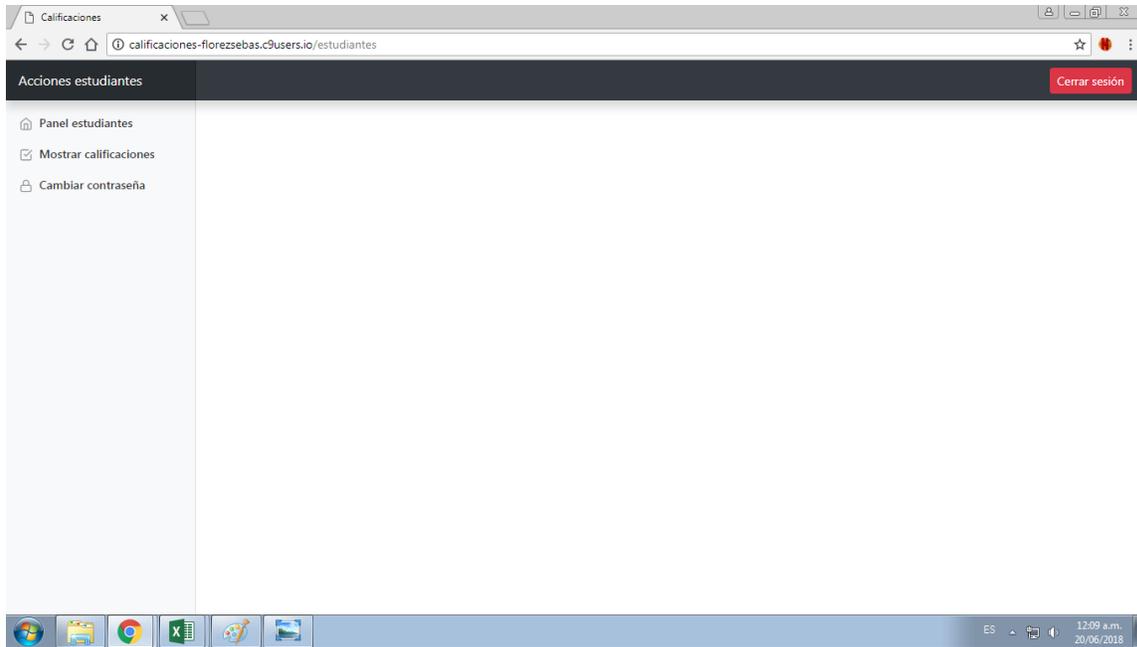


Ilustración 18. Portal estudiante (Elaboración propia)

7.5.4 MÓDULO DE ACUDIENTE

El módulo de acudiente está desarrollado para que los acudientes tengan información académica de forma oportuna de los estudiantes que tiene a su cargo, esta información incluye las calificaciones y adicional a esto, observaciones realizadas por el docente.

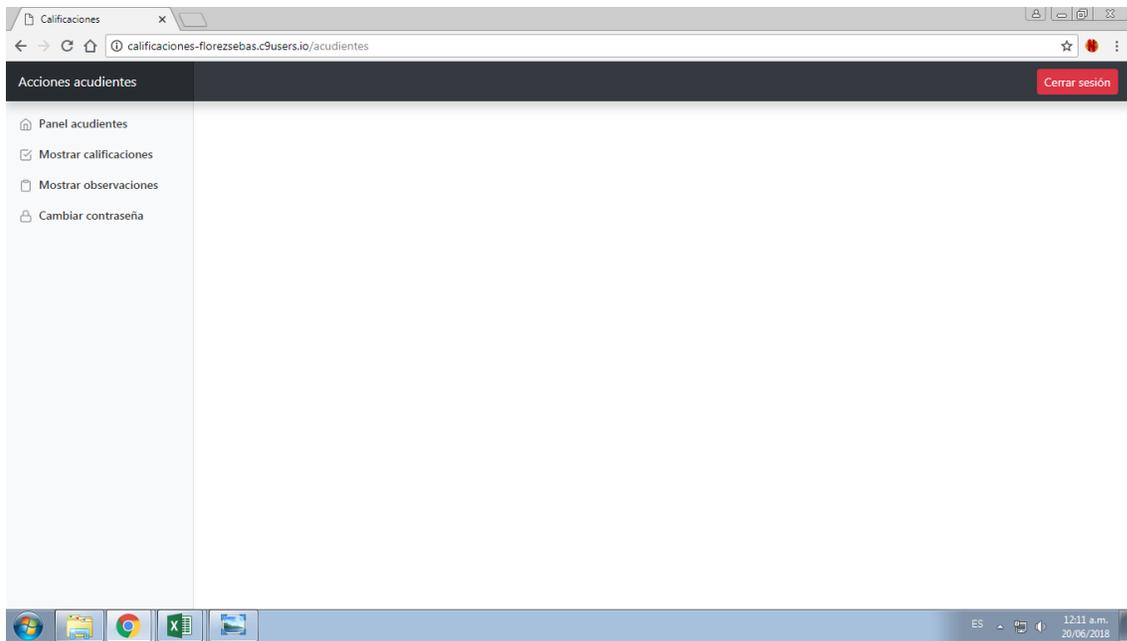


Ilustración 19. Portal acudientes (Elaboración propia)

7.5.5 MÓDULO DE INICIO DE SESIÓN

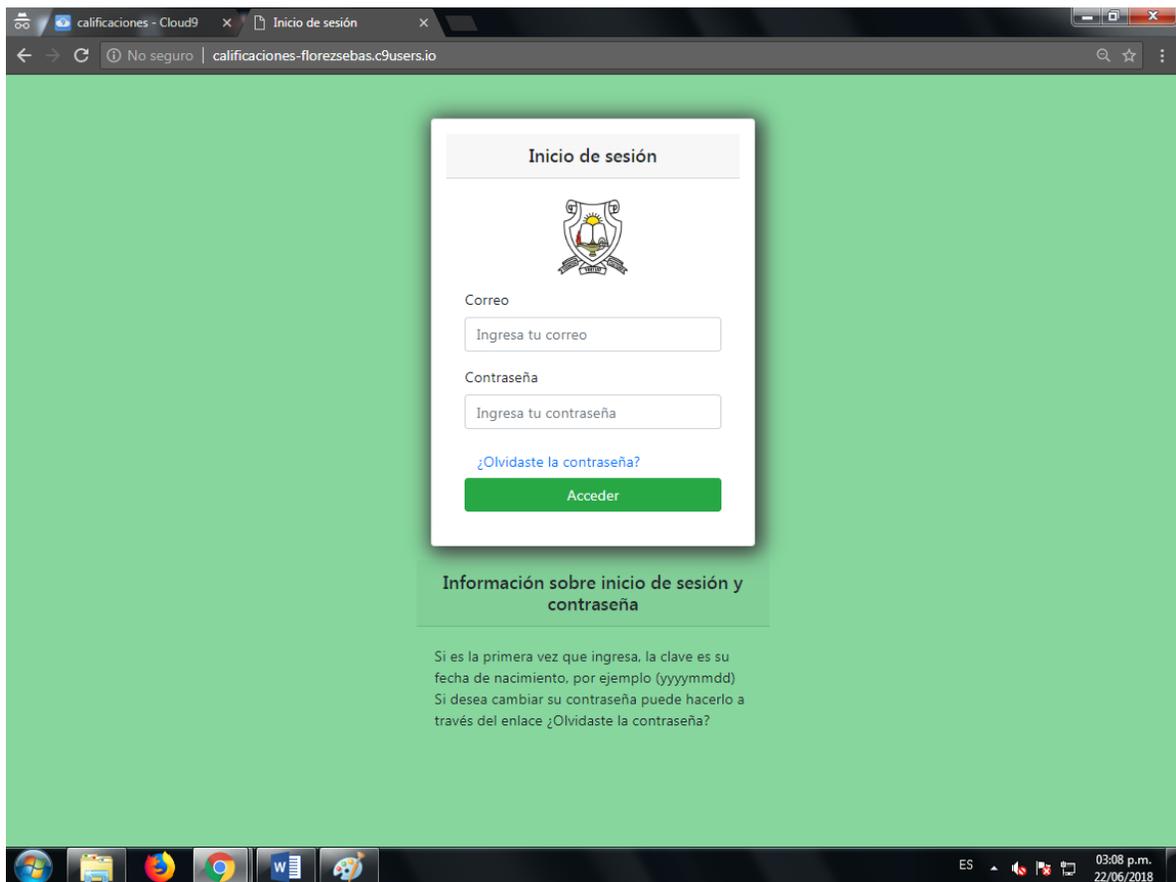


Ilustración 20. Inicio de sesión (Elaboración propia)

7.6 DESPLIEGUE

Para poner en marcha el sitio, se hace a través de cloud9, que es la plataforma que nos permitió desarrollar el prototipo del sistema. Para mayor información sobre el proceso de despliegue ver el anexo.

7.7 PRUEBAS

Es pertinente mencionar primero que durante la etapa de codificación se hicieron paralelamente revisiones al funcionamiento de cada uno de los distintos módulos del sistema y a sus componentes internos a medida que estos fueron creados.

A continuación se detallan los casos de prueba para el prototipo del aplicativo web, donde se toma como referencia los casos de uso propuestos, se prueban los requerimientos funcionales y no funcionales. Casos en los cuales se prueban entradas y se comprueban sus salidas.

7.7.1 ELEMENTOS DE PRUEBA

Los componentes a los que se les realizó pruebas:

- Ingreso a la página
- Inicio de sesión
- Cerrar sesión
- Creación de usuarios
- Editar usuario
- Creación de cursos
- Editar curso
- Creación de actividades
- Editar actividad

- Calificar actividad
- Editar calificación
- Creación de observaciones
- Editar observación
- Consultar calificaciones
- Consultar observaciones

7.7.2 RESULTADOS DE LAS PRUEBAS

- Las pruebas en el inicio de sesión resultaron satisfactorias, cada usuario fue redirigido a su sección correspondiente en el sitio, además se evidenció la protección de las zonas del sitio restringidas para cada usuario.
- Las pruebas de creación de usuarios, cursos, actividades permitió demostrar un informe eficaz en cuanto al flujo de datos entre el prototipo y la base de datos.
- Se validó los formularios de registro de usuarios y la validación de campos desplegando mensajes de error y alertas.
- El prototipo permite la modificación de los usuarios y otras instancias del sistema.

8 CONCLUSIONES

El desarrollo de este proyecto implica que el Colegio Agustín Nieto Caballero de Dosquebradas Risaralda se favorecerá como institución educativa al incrementar la eficiencia académica de los estudiantes, ya que al momento de registrar las notas y observaciones, estarán disponibles a los acudientes, lo que les permitirá estar al tanto de la situación académica de sus hijos y actuar oportunamente.

El modelo de base de datos se desarrolló asegurando cardinalidades tipo 1 -> N lo que garantiza una base de datos rápidos e íntegros, garantizando transacciones rápidas y seguras.

Al ser PHP un lenguaje que oculta código fuente del sistema, brindará seguridad ya que no estará disponible a merced de usuarios maliciosos de la institución que desee provocar daños al sistema.

Laravel es un framework de desarrollo que brinda seguridad en el desarrollo de páginas web que deba contar con sistema tipo login, asegurando así que usuarios no autorizados, ingrese a secciones críticas del sistema y manipule características no autorizadas mediante el uso de middlewares.

Al hacer uso de la metodología de ingeniería del software XP (extreme programming) se pudo llevar a cabo un diseño y desarrollo sólido, rápido y seguro del sistema, en equipo, fortaleciendo los vínculos afectivos del equipo desarrollador así como el sentido de pertenencia hacia el proyecto.

9 RECOMENDACIONES

Capacitar al administrativo, docentes, estudiantes y acudientes en el uso del aplicativo, con el fin de asegurar una rápida adaptabilidad del sistema.

Cambiar contraseñas de cuentas tipo administrador y docentes al menos una vez al mes.

Los estudiantes deberían cambiar inmediatamente la contraseña de su cuenta, una vez hayan iniciado sesión por primera vez.

La adquisición de hosting y de un dominio propio para la operatividad del sistema.

10 BIBLIOGRAFÍA

[1] <http://cds.cern.ch/record/1998446>

[2] <https://www.w3.org/1999/05/WCA-terms/>

[3] <https://searchsqlserver.techtarget.com/definition/database>

[4] <https://searchoracle.techtarget.com/definition/MySQL>

[5] <http://php.net/manual/es/intro-what-is.php>

[6] <http://php.net/manual/es/history.php.php>

[7]

https://books.google.es/books?hl=es&lr=lang_es&id=Q4VL8Tiy7gcC&oi=fnd&pg=PP2&dq=html&ots=Vj59EZ7tuh&sig=7XQAaXqvgonYwyLBtt_IPRYS0CU#v=onepage&q=html&f=false página 12

[8] <https://webdevelopermtv.wordpress.com/frameworks/>

[9]

https://books.google.es/books?hl=es&lr=&id=BWO4CAAQBAJ&oi=fnd&pg=PP1&dq=laravel&ots=8aWaWvALTO&sig=Hdsx1eT8QjL41LWqwvlwAdS_6v4#v=onepage&q=laravel&f=false página 5 y 6