

# UNIVERSIDAD TÉCNICA DEL NORTE



## Facultad de Ingeniería en Ciencias Aplicadas

### Carrera de Ingeniería en Sistemas Computacionales

**APLICACIÓN WEB MÓVIL PARA LA GESTIÓN DE PLANIFICACIÓN CURRICULAR EN LA UNIDAD EDUCATIVA SAN PEDRO PASCUAL DE LA CIUDAD DE IBARRA UTILIZANDO HERRAMIENTAS LIBRES CODEIGNITER Y IONIC 2.**

Trabajo de grado previo a la obtención del título de Ingeniero en Sistemas Computacionales

Autor:

Anthony Patricio Valverde Quispe

Director:

Msc. Daisy Elizabeth Imbaquingo Esparza

Ibarra-Ecuador

2017-2018



## UNIVERSIDAD TÉCNICA DEL NORTE BIBLIOTECA UNIVERSITARIA

### AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

#### 1.- IDENTIFICACION DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO	
<b>CÉDULA DE IDENTIDAD:</b>	1004131916
<b>APELLIDOS Y NOMBRES:</b>	Valverde Quispe Anthony Patricio
<b>DIRECCION</b>	Alpachaca, calle Tucan entre Puyo y Tena
<b>EMAIL</b>	apvalverdeq@utn.edu.ec
<b>TELÉFONO MÓVIL</b>	0969365003

DATOS DE LA OBRA	
<b>TÍTULO:</b>	APLICACIÓN WEB MÓVIL PARA LA GESTIÓN DE PLANIFICACIÓN CURRICULAR EN LA UNIDAD EDUCATIVA SAN PEDRO PASCUAL DE LA CIUDAD DE IBARRA UTILIZANDO HERRAMIENTAS LIBRES CODEIGNITER Y IONIC 2.
<b>AUTOR(ES):</b>	Valverde Quispe Anthony Patricio
<b>FECHA:</b>	20/11/2018
<b>PROGRAMA:</b>	PREGRADO

<b>TITULO POR EL QUE OPTA:</b>	INGENIERO EN SISTEMAS COMPUTACIONALES
<b>ASESOR / DIRECTOR:</b>	MSC. DAISY IMBAQUINGO

## 2.- CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros. Ibarra, a los 20 días del mes de Noviembre de 2018

El autor:



(Firma).....

Nombre: Anthony Patricio Valverde Quispe

Cedula: 1004131916

## **DEDICATORIAS.**

Quiero dedicar este trabajo de titulación de manera especial a mi madre que ha estado a mi lado brindándome su apoyo y sacrificio para que pueda culminar con mis estudios, siendo ella mi motivación que me impulsa a seguir adelante, a mi familia que me ha acompañado en los buenos y malos momentos, brindándome consejos para poder seguir adelante.

Anthony Valverde

## **AGRADECIMIENTOS**

Agradezco a mi familia y amigos que me han acompañado en todo momento brindándome su apoyo y consejos.

Agradezco a mi directora de tesis, Msc. Daisy Imbaquingo quien ha sabido guiarme y aconsejarme durante la realización de este proyecto.

Anthony Valverde



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

Ibarra, 18 de julio del 2018

**CERTIFICACIÓN DIRECTOR DE TESIS**

Por medio del presente yo Msc. Daisy Imbaquingo, certifico que el Sr. Anthony Patricio Valverde Quispe portador de la cédula de identidad Nro. 100413191-6. Ha trabajado en el desarrollo del proyecto de tesis **APLICACIÓN WEB MÓVIL PARA LA GESTIÓN DE PLANIFICACIÓN CURRICULAR EN LA UNIDAD EDUCATIVA SAN PEDRO PASCUAL DE LA CIUDAD DE IBARRA UTILIZANDO HERRAMIENTAS LIBRES CODEIGNITER Y IONIC 2**, previo a la obtención del título de ingeniería en sistemas computacionales, lo cual ha realizado en su totalidad.

Es todo cuanto puedo certificar en honor a la verdad.

---

Msc. Daisy Imbaquingo  
**DIRETOR DE TESIS**



**UNIDAD EDUCATIVA FISCOMISIONAL "SAN PEDRO PASCUAL"  
RELIGIOSOS MERCEDARIOS**

"LIBRES PARA LIBERAR"


IBARRA - ECUADOR

**Ibarra. 16 de Julio 2018.**

La Unidad Educativa Fiscomisional "San Pedro Pascual" certifica que el Trabajo de Grado titulado "APLICACIÓN WEB MÓVIL PARA LA GESTIÓN DE PLANIFICACIÓN CURRICULAR EN LA UNIDAD EDUCATIVA SAN PEDRO PASCUAL DE LA CIUDAD DE IBARRA UTILIZANDO HERRAMIENTAS LIBRES CODEIGNITER Y IONIC 2.", realizado por el sr. Anthony Patricio Valverde Quispe con cédula de ciudadanía 1004131916 ha cumplido con las expectativas requeridas en planificación de acuerdo con lo estipulado en el Ministerio de Educación.

Es cuanto puedo certificar en honor a la verdad.

Atentamente

  
Dr. Luis Enriquez Aldas  
Rector U.E.S.P.P



# ÍNDICE DE CONTENIDO

<b>AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE</b> .....	II
<b>DEDICATORIAS</b> .....	IV
<b>AGRADECIMIENTOS</b> .....	V
<b>ÍNDICE DE CONTENIDO</b> .....	VIII
<b>ÍNDICE DE TABLAS</b> .....	XI
<b>ÍNDICE DE GRÁFICOS</b> .....	XII
<b>Resumen</b> .....	XIV
<b>ABSTRACT</b> .....	XV
<b>Introducción</b> .....	1
<b>CAPÍTULO I</b> .....	6
<b>MARCO TEÓRICO</b> .....	6
1.1. Framework Codelgniter como núcleo del sistema.....	6
1.1.1. Características de Codelgniter .....	7
1.2. Framework Ionic 2 para aplicación de notificaciones .....	8
1.2.1. Características de Ionic 2 .....	9
1.2.2. Diferencias entre Ionic 1 y Ionic 2.....	9
1.2.3. Ventajas y desventajas de Ionic 2 .....	10
1.3. Planificaciones Curriculares como proceso de gestión académica .....	11
1.3.1. Niveles de concreción curricular .....	12
1.3.2. Planificación Curricular Institucional .....	13
1.3.3. Planificación Curricular Anual .....	14
1.3.4. Planificación de Aula .....	15
1.4. Aplicaciones móviles web y aplicaciones híbridas.....	15
1.4.1. Aplicaciones Nativas .....	16
1.4.2. Aplicaciones Web.....	17
1.4.3. Aplicaciones Híbridas .....	18
1.5. Servidor de aplicaciones .....	18
1.5.1. Servidor web Apache .....	19
1.6. Base de Datos MySQL .....	19
1.6.1. Lenguaje de Consultas Estructurado SQL .....	20
1.6.2. Características de MySQL.....	21
1.7. Patrón de diseño – MVC .....	21



1.7.1.	Modelo.....	23
1.7.2.	Vista.....	23
1.7.3.	Controlador.....	23
1.8.	Metodología XP.....	23
1.8.1.	Bases de la programación extrema.....	24
1.8.2.	Principios.....	24
1.8.3.	Prácticas.....	26
<b>CAPÍTULO II.....</b>		<b>29</b>
<b>DESARROLLO.....</b>		<b>29</b>
2.1.	Planificación del proyecto.....	29
2.1.1.	Proceso de gestión de planificaciones curriculares.....	30
2.1.2.	Especificación de requerimientos.....	31
2.1.3.	Tipos de Usuario.....	31
2.1.4.	Historias de Usuario.....	31
2.1.5.	Definición de acrónimos y Abreviaturas.....	33
2.2.	Descripción General.....	34
2.2.1.	Perspectiva del Producto.....	34
2.2.2.	Funcionalidades del Proyecto.....	34
2.2.3.	Restricciones de la Aplicación.....	35
2.2.4.	Requisitos de Interfaz.....	36
2.2.5.	Requisitos de Software.....	36
2.2.6.	Requisitos Funcionales.....	36
2.2.7.	Requisitos No funcionales.....	37
2.3.	Implementación del prototipo.....	38
2.3.1.	Especificación de los casos de uso.....	38
2.3.2.	Casos de Uso de la aplicación.....	41
2.4.	Diseño de la Aplicación.....	47
2.4.1.	Arquitectura del sistema.....	47
2.4.2.	Diagrama de Base de Datos.....	49
2.4.3.	Diagrama de Paquetes.....	50
2.5.	Iteraciones.....	50
2.6.	Primera Iteración.....	50
2.6.1.	Historia de Usuario 1.....	50
2.7.	Segunda Iteración.....	53
2.8.	Tercera Iteración.....	62
2.9.	Cuarta Iteración.....	64

2.10.	Quinta Iteración .....	67
2.11.	Pruebas de la aplicación .....	70
2.11.1.	Pruebas Unitarias .....	70
2.11.2.	Pruebas de Integración .....	71
<b>CAPITULO III</b>	.....	75
<b>IMPLEMENTACIÓN</b>	.....	75
3.1.	Instalación.....	75
3.2	Requisitos. ....	75
3.2.1	Requisitos de servidor.....	75
3.3	Alojamiento .....	75
3.4	Despliegue.....	76
3.5	Matriz de valoración de entrega de documento. ....	76
3.6.	Pruebas de Aceptación .....	76
	CONCLUSIONES.....	79
	RECOMENDACIONES .....	80
	BIBLIOGRAFIA.....	81
<b>Anexos</b>	.....	83

## ÍNDICE DE TABLAS.

Tabla 1 Descripción de módulos .....	4
Tabla 2. Diferencias entre Ionic versión 1 y 2. ....	10
Tabla 3. Ventajas y Desventajas de Ionic 2.....	10
Tabla 4. Niveles de Concreción Curricular.....	13
Tabla 5. Representación las distintas herramientas, lenguajes, formatos y canales de distribución vinculados con los principales sistemas operativos móviles .....	17
Tabla 6. Características de Apache .....	19
Tabla 7. Principios XP .....	25
Tabla 8. Practicas primarias metodología XP.....	26
Tabla 9. Practicas corolarias metodología xp.....	27
Tabla 10. Historia de usuario 01 .....	31
Tabla 11. Historia de usuario 02.....	32
Tabla 12. Historia de usuario 03.....	32
Tabla 13. Historia de usuario 04.....	32
Tabla 14. Historia de usuario 05.....	33
Tabla 15. Historia de usuario 06.....	33
Tabla 16. Funcionalidades del proyecto.....	34
Tabla 17. Funcionalidades del proyecto-Administrador.....	34
Tabla 18. Funcionalidades del proyecto-Rector.....	35
Tabla 19. Funcionalidades del proyecto-Jefe de Área.....	35
Tabla 20. Funcionalidades del proyecto-Docentes.....	35
Tabla 21. Requisitos Funcionales.....	36
Tabla 22.Requisitos No Funcionales .....	37
Tabla 23. Descripción de caso de uso.....	41
Tabla 24. Pruebas Unitarias .....	70
Tabla 25. Pruebas Integración .....	71
Tabla 26. Requisitos Servidor .....	75
Tabla 27 Matriz de valoración .....	76
Tabla 28.Pruebas Aceptación .....	77

## ÍNDICE DE GRÁFICOS

Figura 1 Porcentaje de uso de Computadora (INEC 2017).....	6
Figura 2. Diagrama de flujo de CodeIgniter .....	7
Figura 3 Personas que tienen un teléfono Inteligente (INEC 2017).....	8
Figura 4. Elementos de la planificación curricular institucional.....	14
Figura 5. Top 5 de las BD más populares.....	20
Figura 6. Patrón de diseño MVC.....	22
Figura 7.Patrón de diseño MVC.....	29
Figura 8. Diagrama de flujo del proceso de planificación curricular.....	30
Figura 9. Casos de uso-Actores .....	38
Figura 10. Casos de uso-Administrador .....	39
Figura 11. Casos de uso-Rector .....	40
Figura 12. Casos de uso-Jefe de Área .....	40
Figura 13. Casos de uso-Docente.....	41
Figura 14. Arquitectura de Despliegue .....	48
Figura 15. Diagrama de la Base de Datos .....	49
Figura 16. Diagrama de paquetes .....	50
Figura 17. Ingreso portal administrador.....	51
Figura 18. Administrador Interfaz .....	51
Figura 19. Ingreso porta personal .....	52
Figura 20. Interfaz de personal .....	52
Figura 21. Menú Administrador.....	53
Figura 22. vista personas.....	53
Figura 23. Formulario persona.....	54
Figura 24. Menú Periodo.....	54
Figura 25. Formulario periodo.....	55
Figura 26. Menú Curso.....	55
Figura 27. Formulario Curso.....	55
Figura 28. Menú Paralelo.....	56
Figura 29. Formulario paralelo .....	56
Figura 30. Menú Área de conocimiento.....	56
Figura 31. Formulario Área de Conocimiento.....	57
Figura 32. Menú de usuario.....	57
Figura 33. Formulario Usuario .....	58
Figura 34. Menú editar periodo.....	58
Figura 35. Menú de planificación curricular .....	59
Figura 36. Formulario para agregar curso.....	59
Figura 37. Formulario docente-materia .....	60
Figura 38. Vista malla de materias después de asignar un docente .....	60
Figura 39. Vista jefe de área .....	61
Figura 40. Vista Calendario .....	61
Figura 41. Información del periodo.....	62
Figura 42. Notificación recibo de planificación .....	62
Figura 43. Revisión documento .....	63
Figura 44. Planificación aprobada .....	63
Figura 45. Vista planificaciones Jefe de Área.....	63
Figura 46. Vista circulares .....	64
Figura 47. Vista escribir circular.....	64

Figura 48. Pantalla principal para usuario docentes. ....	65
Figura 49. Acercamiento pantalla principal de usuarios docentes .....	65
Figura 50. Pantalla principal para usuario docentes .....	66
Figura 51. Modal para subir planificación.....	66
Figura 52. Visualización de documentos enviados.....	66
Figura 53. Pantalla de Circulares (Administrador).....	67
Figura 54. Pantalla de Circulares (Docentes).....	67
Figura 55. Visualización completa de un circular (Administrador). ....	68
Figura 56. Redacción de un circular.....	68
Figura 57. Menú Administrador (App Notificaciones).....	68
Figura 58. Inicio de sesión app .....	69
Figura 59. Notificaciones Circular.....	69
Figura 60. Revisión Circular. ....	69

## Resumen

En la actualidad existe mucha demanda de sistemas web como de aplicaciones móviles ya que ayudan a automatizar una gran cantidad de procesos, tareas y servicios asegurando de manera eficaz su cumplimiento, en nuestro país muchas empresas e instituciones también están optando por hacer uso de las nuevas tendencias tecnológicas para poder llevar a cabo de mejor manera sus actividades, dentro del ámbito académico se puede generar una transformación de cambio con la automatización de sus procesos mediante el desarrollo de aplicaciones robustas y estandarizadas, el sistema propuesto en esta investigación está enfocado hacia la gestión de planificaciones curriculares dentro de la institución “San Pedro Pascual”, que busca automatizar y mejorar el proceso académico mediante el uso de software libre como herramienta de desarrollo.

El esquema tecnológico usado para el desarrollo del sistema son: MVC como patrón de arquitectura de software, PHP como lenguaje de programación, CodeIgniter como framework base del sistema, Ionic como framework para el desarrollo de la aplicación móvil, MySQL como base de datos y Bootstrap para el desarrollo de las interfaces, herramientas que ayudaron a obtener un sistema robusto, dinámico, escalable y eficiente con un alto performance en el despliegue de sus funcionalidades de fácil manejo en la gestión, logrando integrar al proceso académico basado en los lineamientos que rige el m<sup>c</sup> Ministerio de Educación y la Institución.

La propuesta tecnológica se realizó bajo la metodología de desarrollo ágil “Programación Extrema”, enfocada a realizar un sistema en lapsos de tiempos cortos, además de ser caracterizado por sus entregas continuas y trabajar de manera conjunta con el cliente, lo que da como resultado un sistema hecho a la medida que cumpla con las expectativas del requeriente haciéndole participe de la solución.

**PALABRAS CLAVE:** Gestión de planificaciones, proceso académico, framework, CodeIgniter, Ionic, sistema web, sistema móvil.

## ABSTRACT

Nowadays, there is a great demand for web systems as well as mobile applications, since they help to automate a large number of processes, tasks and services, effectively ensuring compliance, in our country many companies and institutions are also opting to make use of new ones. technological trends to be able to carry out their activities in a better way, within the academic environment can generate a transformation of change with the automation of their processes through the development of robust and standardized applications, the system proposed in this research is focused on the management of curricular planning within the "San Pedro Pascual" institution, which seeks to automate and improve the academic process through the use of free software as a development tool.

The technological scheme used for the development of the systems are: MVC as a software architecture pattern, PHP as a programming language, CodeIgniter as the base framework of the system, Ionic as a framework for the development of the mobile application, MySQL as a database and Bootstrap for the development of the interfaces, tools that helped to obtain a robust, dynamic, scalable and efficient system with a high performance in the deployment of its functionalities of easy handling in the management, managing to integrate the academic process based on the guidelines that govern the mMinistry of Education and the Institution.

The technological proposal was made under the agile development methodology "Extreme Programming", focused on making a system in short times, in addition to being characterized by its continuous deliveries and working together with the client, which results in a customized system that meets the expectations of the requiriente making him participate in the solution.

**Keywords:** academic process, curricular planning management, framework, CodeIgniter, Ionic, sistema web, sistema movil.

# Introducción

## Antecedentes:

El ministerio de educación a través del acuerdo MINEDUC-ME-2015-00168-A establece “LA NORMATIVA PARA REGULAR LOS PROCESOS DE REGISTRO DE MATRÍCULA, INFORMACIÓN ESTUDIANTIL, PLANIFICACIÓN, EVALUACIÓN EDUCATIVA Y TITULACIÓN EN LAS INSTITUCIONES DEL SISTEMA NACIONAL DE EDUCACIÓN EN EL PORTAL EDUCAR ECUADOR”, donde señala la obligatoriedad a todas las instituciones públicas y privadas la creación de planificaciones curriculares insitucionales, anuales y microcurriculares.

Los formatos y requerimientos de las planificaciones estan establecidos por la ley los mismo que deben ser adoptados por cada una de las instituciones; esta planificaciones son:

PLANIFICACION CURRICULAR PCI contiene la información pedagógica generada en el diagnóstico institucional y es de responsabilidad de las autoridades y docentes de la institución educativa.

PLANIFICACIÓN CURRICULAR ANUAL (PCA), documento que corresponde al segundo nivel de concreción curricular y aporta una visión general de lo que se trabajará durante todo el año escolar; este documento es el resultado del trabajo en equipo de las autoridades y el grupo de docentes de las diferentes áreas (Matemática, Lengua y Literatura, Ciencias Naturales, Ciencias Sociales, Educación Física, Educación Cultural y Artística, Lengua Extranjera), expertos profesionales, y docentes de Educación Inicial.

PLANIFICACIÓN DE AULA (PLANIFICACIÓN MICROCURRICULAR), documento cuyo propósito es desarrollar las unidades de planificación desplegando el currículo en el tercer nivel de concreción; está terminado por el equipo pedagógico institucional de acuerdo a los lineamientos previstos en el PCI; es de uso interno de la institución educativa, por lo tanto los formatos propuestos por la autoridad nacional de educación en relación a esta planificación, son referenciales, ya que las instituciones educativas pueden crear formatos, tomando en cuenta los elementos esenciales: fines, objetivos, contenidos, metodología, recursos y evaluación. (Subsecretaria de Fundamentos Educativos, 2016)

En contexto la planificación curricular debe cumplir la meta de atención a la diversidad ademas de una distribución de responsabilidades en el desarrollo del diseño curricular que comprenda tres niveles de concreción, la misma que deben ser autorizados por la autoridad de la insitución.



El estudio aborda el tema específico para la gestión de la planificación curricular para La Unidad Educativa San Pedro Pascual, se encuentra ubicada en la ciudad de Ibarra, provincia de Imbabura creada en el año de 2004 a través del acuerdo ministerial nro 38 como anexo a la escuela “La Merced” que existía de manera separada en esa época; actualmente dispone de alrededor de 1000 estudiantes distribuidos en los 3 niveles de educación aprobados para la institución: Inicial, Básica y Bachillerato; además, cuenta con una planta docente de la institución es de 59 docentes distribuidos en las diferentes áreas disciplinarias científicas.

### **Situación Actual:**

La Unidad Educativa San Pedro Pascual de la ciudad de Ibarra, responde a las necesidades y requerimientos así como las políticas del Ministerio de Educación del Ecuador (MINEDUC), en la que se establece que las unidades educativas deben responder a una serie de requerimientos siendo la planificación curricular el documento en el que se establecen todos los aprendizajes asociados al proceso de enseñanza aprendizaje articulado con todos los componentes, en la unidad educativa San Pedro pascual, las actividades referidas a este proceso se los realiza a través de documentos electrónicos que se almacenan en una carpeta cloud institucional, dichos documentos son revisados por el responsable académico o autoridad institucional quienes emiten una valoración verbal.

Los inconvenientes que tiene la institución por la forma de cómo lleva el proceso de seguimiento de las planificaciones curriculares se detallan a continuación:

- La búsqueda de los documentos dentro de la carpeta cloud es lenta y tediosa, al tener que buscar manualmente de carpeta en carpeta un documento específico.
- No existe seguridad sobre el control de documentos ya que todo el personal docente como autoridades tienen la misma clave de acceso a la carpeta cloud, provocando un riesgo en el caso de eliminar documentos erróneos.
- En muchas ocasiones los documentos son entregados de manera personal a la autoridad a través de medios electrónicos incluso email dificultando aún más el despliegue en la carpeta cloud
- La información no tiene el criterio de integridad y confidencialidad además al ser un proceso manual es susceptible a errores y manipulación externa.
- El proceso de consolidación de los datos e información se lo lleva de forma manual requieren de tiempo y recursos.

## **Prospectiva:**

Con la implementación del sistema y apoyados con el uso de dispositivos inteligentes móviles que aportan en la toma de decisiones, se pretende perfeccionar el proceso de gestión de las planificaciones curriculares a través una herramienta informática que permita con seguridad subir los documentos por cada uno de los actores del proceso de la planificación curricular de manera ordenada además de conocer en tiempo real el estado de los documentos por notificaciones email y sms para su respectiva aprobación o corrección si se da el caso.

## **Planteamiento del Problema:**

¿Cómo provoca retrasos en el control y entrega de los documentos de planificación curricular por parte de los docentes y coordinadores de área la inexistencia de un sistema de gestión de las planificaciones curriculares en la Unidad Educativa San Pedro Pascual?

## **Objetivos**

### **Objetivo General**

Desarrollar un sistema web y móvil para la gestión de la planificación curricular para la Unidad Educativa San Pedro Pascual utilizando CodeIgniter y Ionic 2.

### **Objetivos Específicos**

- Recopilar las bases teórico-conceptuales que permitan estructurar el proyecto propuesto mediante el análisis de información sobre herramientas de desarrollo de aplicaciones, base de datos, CodeIgniter, Ionic 2, arquitectura de sistemas, ingeniería de software que sustentan el proyecto.
- Realizar el diseño, arquitectura e ingeniería de software de la solución informática propuesta, mediante la metodología XP.
- Implementar un sistema mediante integración de módulos con dispositivos móviles.
- Valorar el cumplimiento de entrega de documentos por el personal autorizado.

## Alcance

Desarrollar un sistema de gestión de planificaciones curriculares para la Unidad Educativa San Pedro Pascual utilizando el framework CodeIgniter para el módulo web y Ionic 2 para el módulo móvil, para ser implementado en la institución.

El sistema estará compuesto por los siguientes módulos y funcionalidades que se describen a continuación en la tabla 1.

Tabla 1 Descripción de módulos

<b>Módulo Web</b>	Gestión de Usuarios	Usuarios Roles. El sistema considera 4 roles: Administrador, Rector, Jefe de Área, Docente
	Gestión de Planificación Curricular	Periodos Cursos Docentes Materias Áreas Gestión de Documentos de Planificación Curricular
	Notificaciones	Circulares Notificaciones Individuales masivas al correo electrónico.
	Reportes	
<b>Módulo Móvil</b>	Gestión de Planificación Curricular	Ver/Descargar Documentos de Planificación Curricular Estado de Documentos Notificaciones masivas al correo electrónico.

Al tratarse de una aplicación web y móvil la aplicación manejará una arquitectura cliente servidor 4 capas base:

- Capa de presentación (Vistas)
- Lógica de negocio (Controladores)
- Acceso a datos (Modelos)
- Base de datos

El modelo vista controlador (MVC), permite la construcción de aplicaciones distribuidas mediante la integración de los modelos, vistas y controladores que son el conjunto de rutinas

y subrutinas que se comunican entre sí para la realización de actividades comunes en el sistema.

### **Justificación**

El proyecto planteado es una necesidad imperativa para la Unidad Educativa “San Pedro Pascual” ya que tiene como objetivo automatizar el proceso que se lleva manualmente de la gestión de las planificaciones curriculares, un software de control que contribuirá a mejorar la calidad de la educación y el desempeño de los docentes.

En la actualidad la mayoría de las instituciones educativas no cuentan con un sistema automatizado ni con las tecnologías necesarias para realizar de mejor manera la gestión de las planificaciones curriculares, por ello es importante que se implementen soluciones tecnológicas como la que presento en este proyecto.

La Unidad Educativa “San Pedro Pascual” tiene procesos y requerimientos con base a lo que el Ministerio de Educación establece para realizar las planificaciones curriculares, por lo que es difícil que se adapte a una solución tecnológica adquirida, por ende, el presente proyecto deberá ajustarse a estos requerimientos y parámetros que exige la institución.

# CAPÍTULO I.

## MARCO TEÓRICO

### 1.1. Framework CodeIgniter como núcleo del sistema.

Los frameworks en la actualidad se han vuelto tendencia al momento de desarrollar aplicaciones ya que son de suma ayuda para el programador, nacen de la necesidad de las personas de utilizar aplicaciones para realizar un sin fin de actividades ya sea desde su ordenador o un teléfono inteligente.

De acuerdo con un estudio realizado en el 2017 por el INEC (Instituto nacional de Estadísticas y Censos) tomando en cuenta un lapso de 5 años, empezando desde el 2012, se ha visto que ha incrementado el uso de ordenadores tales como computadoras de escritorio o laptops en nuestro país, siendo este utilizado desde diferentes medios o lugares, para solventar diferentes necesidades o hacer uso de algún servicio. (INEC, 2017)

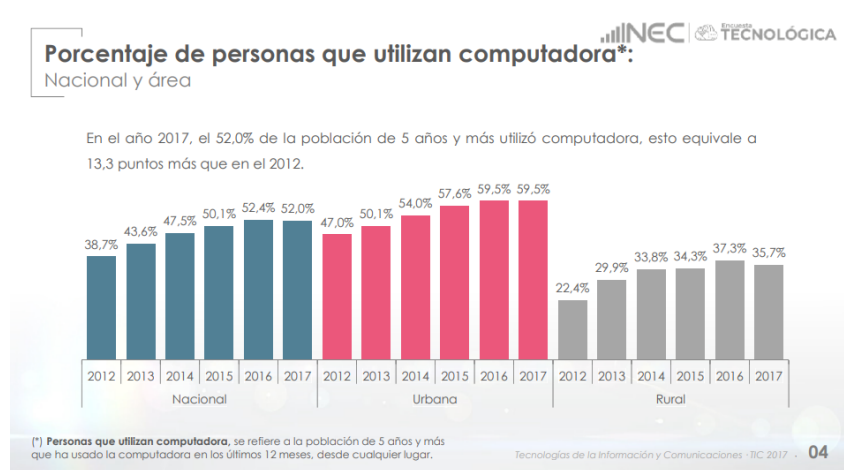


Figura 1 Porcentaje de uso de Computadora (INEC 2017)

Esto quiere decir que más personas van adaptándose al uso de las nuevas tecnologías, al uso de nuevos dispositivos, encontrando una manera más fácil y rápida de realizar diferentes actividades desde cualquier lugar.

Razón por la cual se están automatizando diferentes servicios mediante el uso de sistemas que puedan ser utilizados desde cualquier ordenador o dispositivo móvil. Es por eso que nace la necesidad de desarrollar sistemas de una manera rápida, sin descuidar aspectos de calidad, es por eso que desarrolladores de sistemas optan por utilizar un framework para facilitar un tanto el trabajo y no empezar todo desde cero.

Un framework puede ser definido como un conjunto de librerías que está diseñado para facilitar el desarrollo de un sistema, estos hacen uso de una arquitectura y codificación ya establecida, por eso puede mostrarse como una aplicación genérica a la cual se la puede personalizar obteniendo como resultado un sistema que cumpla con sus respectivas especificaciones. (Navarrete, 2017)

En la actualidad existen muchos framework para desarrollar sistemas web por los cuales se podría optar, uno de ellos es CodeIgniter caracterizado por su facilidad de aprendizaje y desarrollo de sistemas más rápido, ya que permite centrarse en el desarrollo de la aplicación reduciendo la cantidad de código utilizado al mínimo.

CodeIgniter es un framework de código libre que facilita la creación de aplicaciones web usando el lenguaje PHP, contiene varias librerías que sirven para el desarrollo web y maneja una manera específica de codificación propia que exige su estudio y aprendizaje para sacarle el mayor provecho, su estructura maneja el patrón de diseño MVC (Modelo, Vista , Controlador) que es un estándar muy utilizado al momento de desarrollar aplicaciones web, además maneja una interfaz sencilla y un acceso hacia las diferentes librerías muy estructurada, estas facilidades que brinda este framework lo diferencia de otros además de que permite al usuario enfocarse en el desarrollo de su proyecto utilizando la menor cantidad de código posible y tener como resultado una aplicación de calidad. (Alvarez, 2012)

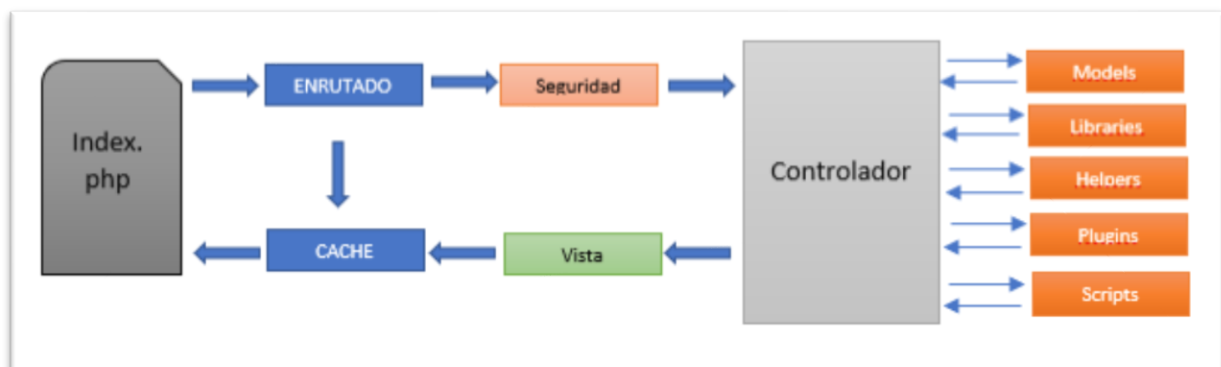


Figura 2. Diagrama de flujo de CodeIgniter

### 1.1.1. Características de CodeIgniter

- a) Una de las características más importantes de este framework es su versatilidad ya que es capaz de trabajar con la mayoría de entornos o servidores.
- b) Es compatible tanto con la versión de PHP 4 y 5 con lo que puede ser utilizado por ciertos servidores un tanto antiguos.

- c) Define una manera específica para desarrollar, está bien puede ser utilizada o no, también se la puede modificar para trabajar de una manera más cómoda.
- d) Su instalación es fácil además su núcleo es ligero con lo que evita que el servidor sufra una sobrecarga al compilar grandes cantidades de código
- e) Cuenta con una documentación a manera de tutorial lo cual nos permita familiarizarnos con el framework su estructura y componentes.

## 1.2. Framework Ionic 2 para aplicación de notificaciones.

Existen frameworks que ayudan a desarrollar tanto aplicaciones web como móviles ya que en la actualidad el uso de un dispositivo móvil inteligente a aumentado ya que mediante estos se pueden realizar transacciones, hacer uso de un servicio, consultas, pagos, etc. El Ecuador también es un país que se adapta al uso de las nuevas tecnologías, de acuerdo con el estudio realizado por el INEC sobre el uso de dispositivos móviles inteligentes, se pudo ver como ha aumentado la tenencia de teléfonos inteligentes en un lapso de 5 años.(INEC, 2017).



Figura 3 Personas que tienen un teléfono Inteligente (INEC 2017)

Es por eso que muchos establecimientos tanto públicos como privados necesitan brindar sus servicios de una manera accesible para el usuario, por lo cual necesitan una aplicación que cumpla con sus especificaciones lo más pronto posible, Razón por la cual los desarrolladores de sistemas necesitan crear aplicaciones móviles que sustenten las necesidades del cliente de una manera rápida, para eso hacen uso de frameworks que permiten desarrollar aplicaciones móviles para las diferentes plataformas que son demanda en la actualidad.

Ionic 2 es un framework basado en cordova y angular, que permite crear aplicaciones móviles multiplataforma de manera rápida y sencilla, este framework utiliza tecnologías web

(Html, Css y Javascript) en sus aplicaciones mediante el uso de una webview el cual es una especie de navegador integrado dentro de una aplicación, este tipo de aplicaciones que hacen uso de esto se los conoce como aplicaciones híbridas.

La facilidad con la que se pueden crear aplicaciones orientadas a diferentes plataformas es lo que vuelve a este framework una opción al momento de desarrollar aplicación ya que básicamente con un mismo código nosotros podemos generar aplicaciones para Windows, Android e IOS. (Phan, 2016)

### **1.2.1. Características de Ionic 2.**

Ionic 2 maneja una mejora en su interfaz de línea de comandos, abarcando muchas más tareas al ejecutar un comando, esto hace posible que al ejecutar pocas líneas de comandos ya obtengamos una aplicación ya funcional además de configurada y estructurada, esto hace que este framework sea preferido por nuevos usuarios.

Maneja una sintaxis de codificación más limpia y simplificada obteniendo un código más limpio y fácil de usar.

Mediante cordova genera la codificación necesaria para obtener una aplicación multiplataforma.

Maneja una mejor manipulación del DOM (Modelo de objetos del documento) basado en angular 2, el cual a su vez tiene un mejor rendimiento en el manejo de DOM en términos de tiempo, esta ventaja es bien aprovechada por ionic 2.

Posee una compilación anticipada AoT (Ahead of Time Compilation), esto permite que en lugar de compilar la plantilla sobre la marcha mediante el navegador se pre-compile en un paso de compilación, esto ayuda a que el paquete sea más pequeño y liviano para descargar además de que la aplicación se iniciara más rápido.

### **1.2.2. Diferencias entre Ionic 1 y Ionic 2**

Las diferencias entre la primera versión de ionic y su segunda entrega son muy notables, puntos como rendimiento diseño y adaptabilidad fueron tomados en cuenta al desarrollar la segunda versión, a continuación, algunas de las diferencias más notables entre cada versión (RENAUX, 2017) :



Tabla 2. Diferencias entre Ionic versión 1 y 2.

<b>Diferencias entre Ionic 1 y 2</b>	
<b>Ionic1</b>	<b>Ionic2</b>
La apariencia de desarrollo de una aplicación era igual para cada plataforma móvil (Android y IOS).	Se obtiene un diseño de aplicación propio de la plataforma móvil en la que se desarrolla
El desarrollo del framework está basado en angular 1	El desarrollo del framework está basado en angular 2
El desarrollador podía manejar de cualquier manera la estructura de su aplicación	Posee una mejor estructura y organización obligando al desarrollador a apegarse a una manera de desarrollo basada en buenas prácticas de manera estricta

### 1.2.3. Ventajas y desventajas de Ionic 2.

Como todo framework que existe en la actualidad tiene sus ventajas y desventajas que influyen en la decisión de un desarrollador al momento de escoger una herramienta para desarrollar una aplicación, características por las cuales algunos frameworks son mejores que otros en aspectos específicos como lenguaje de programación, plataformas de desarrollo, arquitecturas, patrones etc. En el siguiente cuadro se describirán algunas de las ventajas y desventajas que posee la segunda versión del framework Ionic.

Tabla 3. Ventajas y Desventajas de Ionic 2.

Ventajas	Desventajas
Es multiplataforma, con un mismo código podemos generar aplicaciones para Windows, IOS y Android, reduciendo de esta manera el tiempo y coste en el desarrollo de una aplicación.	Utiliza una webview y no los elementos nativos de la plataforma lo que resultara en una menor fluidez de la aplicación al momento realizar ciertas funciones

---

Si se dispone de conocimientos previos en el desarrollo a nivel de front end facilita de mejor manera el aprendizaje del framework.	El rendimiento de la aplicación puede variar dependiendo de los recursos que utiliza en las diferentes plataformas
---	--

---

Dispone de muchos componentes con los que se puede crear una aplicación de apariencia profesional sin necesidad de ser un diseñador. <b>(Revilla, 2017)</b>	Encontrar información o soporte puede tener algo de dificultad ya que Ionic es un framework que va cambiando y mejorando ciertas características.
---	---

---

Al ser una aplicación híbrida siempre estará el uso de una webview para poder ser ejecutado en las diferentes plataformas esto radica en varias ventajas y desventajas para los desarrolladores aunque con la capacidad de los nuevos dispositivos existentes y la mejora entre el rendimiento de una webview en una aplicación en las nuevas versiones de sistemas operativos puede hacer posible que el funcionamiento de una aplicación nativa y una híbrida sea similar, una aplicación bien desarrollada con este framework será similar a una nativa independientemente de la plataforma. (Revilla, 2017)

### **1.3. Planificaciones Curriculares como proceso de gestión académica.**

Para asegurar el aprendizaje y enseñanza dentro de las actividades estudiantiles la institución educativa debe cumplir con la realización de varios procesos académicos estipulados dentro de la constitución y dentro de la Institución.

Un proceso académico se lo puede definir como una serie de actividades repetitivas que tiene como finalidad cumplir con los lineamientos definidos por la institución o por el Ministerio de Educación los cuales permiten desarrollar la actividad académica de una manera organizada cumpliendo con los objetivos, misión y visión planteados por la institución asegurando así su producto final que en este caso son los estudiantes. (Ana Flores, 2014)

Uno de los procesos académicos con los cuales tiene que cumplir la institución es la gestión de planificaciones curriculares el cual consiste en enviar a las autoridades documentos de planificación, en las que constan todas las actividades que se llevarán en el

transcurso del periodo estudiantil, tales planificaciones son realizadas por petición del Ministerio de Educación y están clasificadas dentro de 3 niveles de concreción curricular estas son las planificaciones macrocurriculares, mesocurriculares y microcurriculares, documentos que son realizados tanto por docentes como personal administrativo de la institución.

### **1.3.1. Niveles de concreción curricular**

Comprenden 3 niveles de concreción, están enfocados hacia lo que señala el artículo 9 del Acuerdo Ministerial N.º ME-2015-00168-A de 01-12-2015 en el cual señala como objetivo que la actividad de planificación curricular debe considerar a la atención de la diversidad dentro de lo que es el marco legal educativo (MINEDUC, 2016), a continuación se detallan cada uno de los niveles:

#### **a) Primer Nivel**

Este corresponde a la planificación macrocurricular, este nivel se centra en determinar los objetivos, destrezas, perfil y criterios e indicadores de evaluación que son obligatorios a nivel nacional, es elaborado por docentes, pedagogos, expertos en las diferentes áreas de conocimiento entre otros. Este documento es flexible lo cual permite a las instituciones educativas a partir de lo establecido definir los contenidos de acuerdo con la realidad de su establecimiento y comunidad. (MINEDUC, 2016)

#### **b) Segundo Nivel**

Corresponde a la planificación mesocurricular, dentro de este están comprendidos dos tipos de diseños, la planificación curricular Institucional (PCI) y la Planificación Curricular Anual (PCA), es elaborado de manera conjunta tanto por docentes como autoridades de la institución y responden a lineamientos institucionales como también a la pertinencia cultural de los pueblos y nacionalidades indígenas. (MINEDUC, 2016)

#### **c) Tercer Nivel**

Son documentos curriculares basados en el segundo nivel de concreción y corresponden a la planificación microcurricular, son elaborados por los docentes, es a nivel de aula por ende debe responder a las necesidades de los estudiantes de los diferentes grados o cursos. (MINEDUC, 2016)

Tabla 4. Niveles de Concreción Curricular  
Fuente: (MINEDUC, 2016)

<b>1er nivel</b>	<b>2do nivel</b>		<b>3er nivel</b>
Macro Ministerio de Educación	Meso Instituciones Educativas		Micro Docentes
Currículo nacional obligatorio	Currículo Institucional		Currículo de aula
	Proyecto curricular institucional	Plan curricular anual	Planificaciones de aula. Adaptaciones curriculares (individuales y grupales)
Prescriptivo	<b>Flexible</b>		<b>Flexible</b>

### 1.3.2. Planificación Curricular Institucional.

Es parte del proyecto educativo estudiantil, se centra en la gestión del aprendizaje y tiene una duración de 4 años antes de ser modificado, su elaboración es responsabilidad de autoridades y docentes y tiene la siguiente lógica de construcción (MINEDUC, 2016):

- a) Análisis del currículo nacional: En este paso tiene como objetivo determinar todo lo relacionado hacia al aprendizaje básico enfocado hacia la institución, se toma en cuenta lo que son los objetivos, contenidos, metodología, se examina el perfil, propuestos en el currículo nacional.
- b) Análisis del diagnóstico Institucional: Al ser parte del Proyecto Educativo Institucional se lo debe analizar desde 3 puntos:
  - a. Problemas pedagógicos detectados.
  - b. Factores que influyen en la problemático y soluciones
  - c. Delimitaciones de las necesidades de aprendizaje que deben ser adaptadas al momento de plantear el pensum de estudios.
- c) Delimitación de lineamientos: Después de haber realizado el análisis de los puntos anteriores se procede a fijar los lineamientos de metodología, carga horaria, evaluación entre otros.

Estos puntos sirven como base para realizar el planteamiento de los elementos curriculares, tales elementos son los que se encuentran en la figura 4.

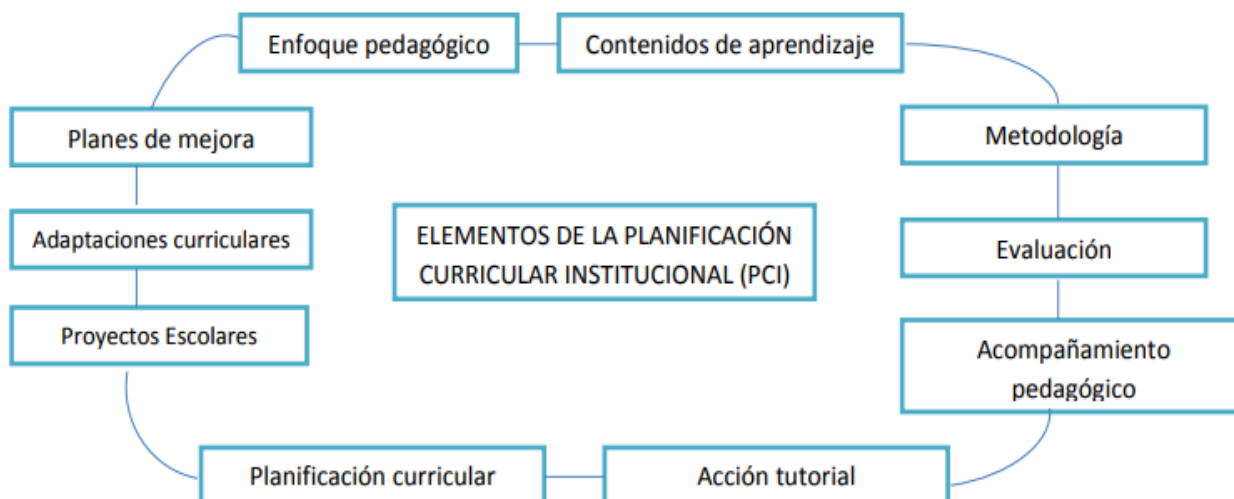


Figura 4. Elementos de la planificación curricular institucional  
Fuente: (MINEDUC, 2016)

### 1.3.3. Planificación Curricular Anual

Este documento se encuentra en el segundo nivel de concreción curricular en donde se verá todo lo que se tratara en el año escolar, es elaborado por todos los docentes de las diferentes áreas y las respectivas autoridades. Esta planificación está compuesta por 8 secciones y cada sección cubre aspectos específicos, a continuación, se detallan las secciones y los aspectos que cubre. (MINEDUC, 2016)

- a) Datos informativos: Todo lo que respecta a información institucional y también abarca los siguientes puntos:
  - a. Área: Áreas que corresponden al currículo de básica y bachillerato
  - b. Asignatura: Asignaturas y profesional que conforma el área.
  - c. Grado/cursó: Se debe especificar el grado o curso de acuerdo con la planificación planteada.
- b) Tiempo: Información relacionada con la carga horaria.
- c) Objetivos generales: Se especifican objetivos del área como de grado o curso.
- d) Ejes transversales/valores: valores que son determinados por la institución en los que se toma en cuenta principios del buen vivir, identidad, misión y contexto institucional.
- e) Desarrollo de unidades de planificación: Aquí se ve una visión general de lo que se trabajara en todo el año escolar en lo que respecta a unidades de estudio.
- f) Recursos: Materiales que se utilizaran para desarrollar las unidades.
- g) Planes de mejora: Actividades que sirven para operativizar lo planteado en el proyecto educativo institucional.
- h) Observaciones: Sugerencias que pueden mejorar el desarrollo de la planificación.

### **1.3.4. Planificación de Aula**

Este documento pertenece al tercer nivel de concreción curricular, es de uso interno así que su formato es elaborado por la institución educativa, además de ser un documento flexible puede ser modificado de acuerdo a las necesidades de la institución, pero debe tomar en cuenta algunos puntos esenciales como los fines que tiene la institución con sus estudiantes, el perfil de salida que se verá reflejado en los estudiantes, objetivos de aprendizaje que se deberán cumplir dependiendo del nivel de educación (Educación Inicial, Educación General Básica, Bachillerato General Unificado), contenidos que se tratarán con los estudiantes, metodología de enseñanzas, recursos y evaluación.

Este documento es elaborado por cada docente encargado de los diferentes grupos de estudiantes en las áreas de básica superior y bachillerato y es revisada por la persona encargada de las áreas de conocimiento o bien por la máxima autoridad de la Institución quienes darán su aprobación para proceder con lo que este planteado en la planificación con los respectivos estudiantes. (MINEDUC, 2016)

### **1.4. Aplicaciones móviles web y aplicaciones híbridas.**

Los dispositivos móviles en la actualidad forman parte del diario vivir de las personas, cada vez son más las personas que acceden a un teléfono inteligente para poder utilizar y aprovechar de mejor manera los servicios que estos dispositivos nos ofrecen, desde mantenernos comunicados con las personas acortando distancias a través de llamadas hasta realizar transacciones bancarias desde cualquier parte que el usuario se encuentre son claros ejemplos de las cosas que podemos hacer con un dispositivo móvil.

Debido al gran consumo de estos dispositivos nace la necesidad de crear aplicaciones móviles como una manera de adaptarse más a la realidad tecnológica del mundo pero el desarrollo de estas es complicado por la gran variedad de dispositivos móviles existen incluso hasta ahora el crear aplicaciones radica en un gran desafío debido a las especificaciones que cada dispositivo tiene, son tantos los aspectos que hay que tomar en cuenta al momento de crear una aplicación móvil, aspectos como el sistema operativo, versiones del sistema, tamaño de pantalla, capacidad de procesamiento, si cuenta con cámara, GPS entre otros; son algunas de las cosas que hay que tener en cuenta al momento de desarrollar una aplicación móvil. (Garita-Araya, 2013)

Ya que existe un gran costo el crear una aplicación para las diferentes plataformas los desarrolladores tuvieron que encontrar una mejor manera para desarrollar aplicaciones

móviles, la solución fue crear una única aplicación que pueda ser utilizada en las diferentes plataformas, de esta manera el costo y tiempo de desarrollo se reduce de una manera muy notable, y así es como nacieron las aplicaciones multiplataforma.

En lo que respecta a aplicaciones móviles podemos clasificarlas en tres, tenemos a las aplicaciones nativas que son aplicaciones creadas para una plataforma específica y tenemos las aplicaciones web e híbridas que son aplicaciones multiplataforma capaces de funcionar en cualquier plataforma. (Garita-Araya, 2013)

#### **1.4.1. Aplicaciones Nativas**

Las aplicaciones nativas son aquellas que fueron desarrolladas para una plataforma específica, en este tipo de aplicaciones se considera el tipo de dispositivo, su sistema operativo y su versión del sistema.

Como son aplicaciones que funcionan en una plataforma específica se puede interactuar con todas las capacidades que el dispositivo posea, la cámara, el calendario, agenda, acelerómetro entre otras; esta sería una de las ventajas que nos da el programar una aplicación nativa, aunque el costo tiempo y esfuerzo de desarrollo se han altas ya que se programa en un lenguaje específico y si se quiere realizar una aplicación para diferentes plataformas radica en programar la misma aplicación en diferentes lenguajes por ende es más tiempo, más esfuerzo y más costo de desarrollo. (Delía, Galdámez, Thomas, & Pesado, 2013)

Este tipo de aplicaciones tienen archivos ejecutables que se descargan y se almacenan localmente, la instalación no suele radicar en mucho problema basta con descargar la aplicación desde la tienda de aplicaciones que la plataforma ofrece, como por ejemplo AppStore de Apple, Google Play de Android entre otros, después de instalada la aplicación esta se conecta con el sistema operativo con la capacidad de acceder a todos los servicios que el dispositivo tiene, esto sin la necesidad de algún intermediario o contenedor que lo haga funcionar. (IBM, 2012)

Para el desarrollo de una aplicación móvil nativa se debe escribir primero el código fuente en un lenguaje propio para la plataforma además de agregar el estilo visual el audio, animaciones dependiendo de lo que la aplicación requiera mediante el uso de herramientas disponibles para la plataforma. A estas herramientas y otros archivos se los conoce como SDK (software development kit) del sistema operativo móvil. Para un mejor entendimiento en

la tabla 5 se muestra diferencias que cada sistema operativo móvil tiene en términos de lenguaje, herramientas, formato y tienda.

Tabla 5. Representación las distintas herramientas, lenguajes, formatos y canales de distribución vinculados con los principales sistemas operativos móviles  
Fuente: (IBM, 2012)

	Apple iOS	Android	Blackberry OS	Windows Phone
Lenguajes	Objective-C, C, C++	Java (algunos C, C++)	Java	C#, VB NET, etObc.
Herramientas	Xcode	Android SDK	BB Java Eclipse Plug-in	Visual Studio, Windows Phone
Formato	.app	.apk	.cod	.xap
Tiendas	Apple App Store	Google Play	Blackberry App World	Windows Phone Marketplace

#### 1.4.2. Aplicaciones Web

Este tipo de aplicaciones son ejecutadas simplemente desde cualquier navegador y en la actualidad los dispositivos móviles cuentan con navegadores capaces de soportar las últimas versiones de html, css y javascript. Como son aplicaciones que se pueden acceder solo desde un navegador no requiere de instalación alguna basta con tener acceso a internet para utilizarlas además como los cambios o actualizaciones que se realicen son aplicadas directamente en el servidor, son vistos de manera rápida en el navegador. Su principal ventaja radica en que trabaja de manera independiente al sistema operativo en el que se lo utilice solo hace uso de internet y del navegador. Pero su rendimiento y calidad disminuye por varios factores como velocidad de internet o el hecho de poder acceder a varios elementos del dispositivo. (Delía, Galdámez, Thomas, & Pesado, 2013)

Pese a las limitaciones que una aplicación web tiene en la parte móvil debido a sus limitaciones de uso de componentes de dispositivo, suele ser una buena opción a la hora de desarrollar una aplicación móvil además debido a los grandes avances de html la brecha entre el uso de todos los componentes que la aplicación puede utilizar del dispositivo es más corta, si bien el navegador de un dispositivo es una aplicación nativa que tiene acceso a los componentes del mismo, para una aplicación web mucho de estos componentes no están disponibles o solo pueden ser usados de manera parcial siendo una limitación al potencial que puede obtener un aplicación web móvil, aunque esto puede cambiar en un futuro gracias



a los avances que se podrían obtener en estudios acerca de cómo mejorar las funcionalidades por parte de las tecnologías web. (IBM, 2012)

### **1.4.3. Aplicaciones Híbridas**

Las aplicaciones híbridas utilizan lo mejor de las aplicaciones nativas y web. Utilizando tecnologías multiplataforma como HTML, CSS y Javascript además de poder acceder a una buena parte de las capacidades específicas que los dispositivos ofrecen. Estas aplicaciones son desarrolladas mediante tecnologías web y ejecutadas dentro de un contenedor web en el dispositivo móvil.

Alguna de sus ventajas son la reutilización de código para las diferentes plataformas, el acceso a varios de los componentes del dispositivo y la posibilidad de distribución en las tiendas de las diferentes plataformas. Pero estas aplicaciones suelen utilizar una misma interfaz para todas las plataformas así que no lucirá como una aplicación nativa y su rendimiento puede ser un poco más lenta en comparación con una aplicación nativa. (Delía, Galdámez, Thomas, & Pesado, 2013)

Para el desarrollo de este tipo de aplicaciones los programadores suelen escribir gran parte de la aplicación haciendo uso de tecnologías web y solo cuando lo necesitan hacen uso de acceso hacia los componentes del dispositivo. Esta técnica hace que la aplicación solo pueda ser utilizada mientras el dispositivo tenga acceso a internet quitando las posibilidades de uso offline pero también puede mejorar el uso de aplicación ya que hace uso de recursos HTML lo que mejora la flexibilidad de la aplicación además de poder acceder a la aplicación de manera remota. (IBM, 2012)

## **1.5. Servidor de aplicaciones**

Los servidores de aplicaciones son tecnologías que nos permiten desplegar aplicaciones de forma rápida y confiable de manera independiente del cliente, en el podemos realizar las pruebas que sean necesarias del sistema que se está desarrollando.

Hay una gran cantidad de servidores de aplicaciones cada uno con características que los diferencian entre ellos entre estos servidores tenemos a apache que es uno de los más populares a la hora de escoger un servidor ya que satisface muchos aspectos que un servidor web debe satisfacer por ejemplo: Es multiplataforma, posee una gran variedad de paquetes lo cual nos permite trabajar con una gran variedad de lenguajes de programación web, así como varios gestores de base de datos SQL, también nos permite realizar transacciones seguras y soporte para hosts virtuales.

### 1.5.1. Servidor web Apache

Apache es uno de los servidores más utilizados a escala mundial, es un servidor de licencia libre por lo que cualquier persona puede hacer uso de él, basta con descargarlo e instalarlo, además puede ser configurado de manera que se adapte a las necesidades del usuario, su primerio aparición fue en abril de 1995 y en diciembre de ese mismo año ya apareció la versión 1.0, realizada por Apache Group una empresa sin ánimo de lucro que ha ido creciendo poco trabajando a través de internet, aunque el desarrollo de este servidor no está limitado a las personas que trabajan en este grupo, cualquier persona puede trabajar y mejorar este servidor aunque el grupo tendrá la decisión de si utilizar alguna mejora dada por un usuario, siendo así que miles de personas han trabajado en la mejora de este servidor. (Talón, 2016)

Entre algunas de las características de este servidor tenemos las siguientes:

Tabla 6. Características de Apache

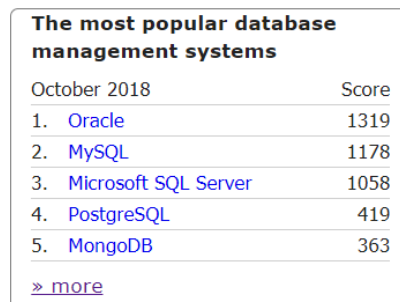
<b>Características de Apache</b>
Trabaja con la versión del protocolo HTTP siendo totalmente compatible con la mayoría de aplicaciones.
Es muy simple a la hora de configurar, solo cuenta con tres ficheros en texto plano que pueden en el que se pueden modificar los parámetros del servidor basta con tener un editor de texto para configurar.
Admite servidores virtuales, puede trabajar utilizando una dirección IP o un nombre virtual.
Puede convertirse en un servidor proxy
Registra el estado del servidor y puede ser visualizado por el usuario

### 1.6. Base de Datos MySQL

Una base de datos se la puede definir como un conjunto de datos estructurados que facilita la gestión de información, estas pueden ser desde archivos planos o documentos Xml hasta

gestores de datos que manejan de mejor forma el grado de estructuración de la información como MySQL, PostgreSQL, Sql Server entre otras consideradas ya como sistemas de gestión de bases de datos.

En la actualidad existe muchos gestores de bases de datos para escoger, independientemente de ser pagadas o de uso libre cada uno cuenta con características que las hacen diferentes del resto. De acuerdo a un estudio realizado por solid IT, una empresa dedicada a desarrollar software y brindar consultoría y entrenamiento acerca de bases de datos, tomando en cuenta a más de 300 tipos de bases de datos y varios aspectos que hacen de los gestores de bases de datos mucho mejor que otros en comparación, llego a la conclusión de que MySql se encuentra en segundo lugar seguido por Oracle lo que hace que MySql sea una opción confiable a la hora de escoger una base de datos de tipo open source, estudio que fue publicado en su página de internet db-engines.com (IT, 2018)



The most popular database management systems	
October 2018	Score
1. <a href="#">Oracle</a>	1319
2. <a href="#">MySQL</a>	1178
3. <a href="#">Microsoft SQL Server</a>	1058
4. <a href="#">PostgreSQL</a>	419
5. <a href="#">MongoDB</a>	363
<a href="#">» more</a>	

Figura 5. Top 5 de las BD más populares

MySQL es un sistema de gestión de bases de datos uno de los más populares a nivel mundial, fue desarrollado por MYSQL AB en Suecia en 1995, MYSQL AB es una compañía de segunda generación dedicada a desarrollar software para la base de datos MySQL, el objetivo principal al crear la base de datos fue el de cumplir con el estándar de SQL (Lenguaje de consultas estructurado) tomando en cuenta factores como la velocidad, fiabilidad o usabilidad y como cualquier gestor de base de datos nos permite administrar la información, actualizar y eliminar como sea necesario. (Torres, 2014)

### 1.6.1. Lenguaje de Consultas Estructurado SQL

SQL, es la base de las aplicaciones de MySQL es el lenguaje mediante el cual podemos realizar consultas a la base de datos, pero no es la única función que tiene, entre algunas de sus funciones están las siguientes:

- Crear Bases de datos
- Crear y gestionar tablas en una base de datos.

- Gestionar datos
- Agrupar datos
- Crear índices y claves.

Ya que la norma de SQL no se impuso de una manera tan estandarizada, existen diferentes dialectos de SQL en los diferentes sistemas de gestión de base de datos esto quiere decir que los comandos utilizados por MySQL diferirán con los comandos utilizados por otras bases de datos SQL. (Torres, 2014)

### **1.6.2. Características de MySQL**

Como ya se mencionó MySQL es un sistema de gestión de base de datos, escrito en el lenguaje C y C++, el cual nos permite gestionar la información de una manera estructurada, pero además de eso es relacional, una base de datos relacional es aquella que maneja la información mediante tablas lo que mejora la velocidad y flexibilidad en el manejo de los datos.

Es Open Source cuenta con una licencia GNU (General Public License) lo que quiere decir que cualquier persona pueda hacer uso de esta y modificarlo para adaptarlo a sus necesidades, se lo encuentra fácilmente en internet no tiene costo alguno y solo es cuestión de descargarlo.

Su forma de trabajo es la de un entorno de cliente/Servidor o incrustados esto consiste en un servidor que en este caso es un SQL multi-threaded y múltiples backends que vendrían a ser programas, herramientas administrativas bibliotecas cliente entre otros.

A pesar de que fue desarrollado originalmente solo para manejar bases de datos de una manera más rápida en comparación con las que existían en su tiempo MySQL ha ido mejorando año tras año mejorando su rendimiento, ofreciendo en la actualidad una mejor conectividad, velocidad y seguridad justificando el porqué de su popularidad entre otras bases de datos.

### **1.7. Patrón de diseño – MVC**

Los patrones de diseño dentro de lo que es la informática son modelos que pueden ser aplicados al momento de realizar un sistema, estos modelos están basados en soluciones a problemas comunes en un contexto específico que los programadores se encuentran al momento de desarrollar software, este término tiene su origen en la arquitectura cuando

Cristopher Alexander intento crear patrones específicos para diseños de edificios, extrayendo partes comunes de buenos diseños con la finalidad de volver a utilizarse.

Cristopher Alexander tenía la siguiente definición de lo que un patrón es:

“Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, para describir después el núcleo de la solución a ese problema, de tal manera que esa solución pueda ser usada más de un millón de veces sin hacerlo ni siquiera dos veces de la misma forma”. (Erich, Richard, Johnson, & Vlissides, 2012).

Todo este conocimiento fue utilizado ya en el ámbito de desarrollo de software, lo que dio como resultado varios patrones de diseños en resolución a problemas comunes a los que se enfrentaban los programadores, la utilización de estos patrones en la actualidad demuestra un gran conocimiento y madurez por parte del programador.

Uno de los patrones de diseño al desarrollar sistemas web es el MVC (Modelo, Vista, Controlador) que consta de tres capas en las que se separan la lógica del sistema, la interfaz y el acceso a datos del sistema, en la figura 6 se muestra el funcionamiento de este patrón.

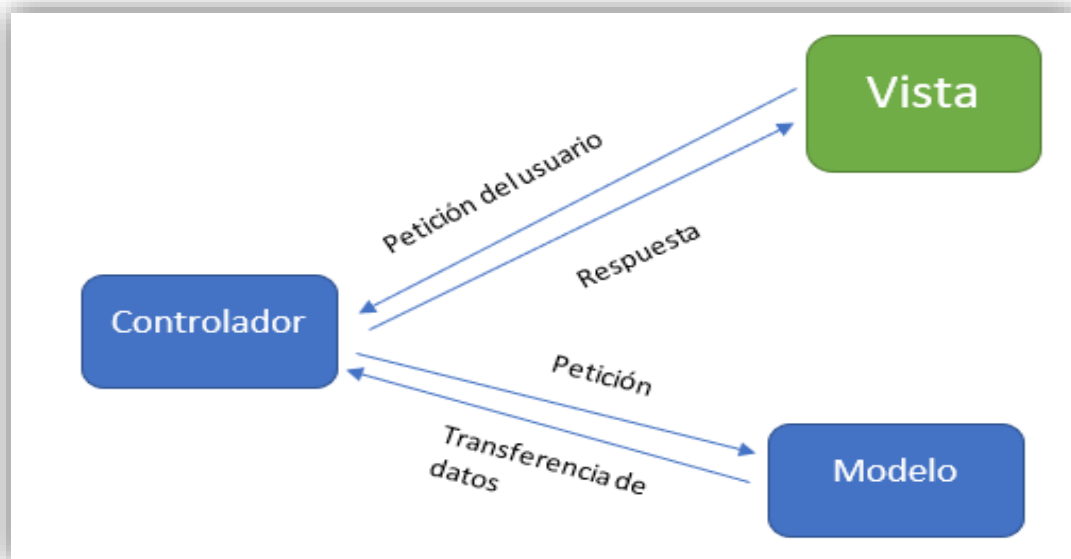


Figura 6. Patrón de diseño MVC

Como se observa en la figura 6 cada capa cumple con una función específica y se comunican a través de peticiones hacia el controlador que es la capa que interacciona con la capa de la viste y el modelo, la manera de funcionamiento del patrón empieza por una petición por parte del usuario la petición es mandada hacia la capa del controlador que realiza las

operaciones necesarias y peticiones de información a la capa del modelo devolviendo la respuesta hacia la capa de la vista.

### **1.7.1. Modelo.**

En esta capa se encuentra todo lo relacionado a las reglas del negocio, aquí se establece la conexión con la base datos para la extracción y actualización de los datos, esta capa es responsable de recibir la petición por parte del controlador realizar las funciones necesarias y extraer o modificar la información en la base de datos. (Yenisleidy & Yanette, 2012)

### **1.7.2. Vista**

Esta capa es la responsable de enviar las peticiones por parte del usuario hacia el controlador, también muestra la información que viene desde la capa del modelo al usuario, en esta capa se verán reflejados todos los cambios que se den en el modelo. (Yenisleidy & Yanette, 2012)

### **1.7.3. Controlador**

Es el encargado de interactuar con la capa de vista y modelo, en esta capa se encuentra la gestión de eventos, recibe peticiones por parte de la vista y manda una respuesta a la misma después de haber enviado una petición a la capa de modelo y haber recibido su respuesta. (Yenisleidy & Yanette, 2012)

## **1.8. Metodología XP**

Al momento de poner en marcha un proyecto existen varios métodos para el desarrollo y gestión por el cual se puede guiar para tener como resultado un proyecto exitoso, en lo que a desarrollo de software se trata existen varias metodologías que pueden ser utilizadas como guías, una de estas son las metodologías ágiles, estas metodologías se centran en los posibles cambios que puedan surgir durante el desarrollo del proyecto y se adaptan a ellos, evolucionan de manera conjunta con el software.

La metodología XP (Extreme Programing) que en español sería programación extrema es una metodología ágil utilizada para proyectos de corto plazo, está estrictamente ligado a ciertas reglas enfocadas para satisfacer las necesidades del cliente y lograr como resultado un producto final de buena calidad en poco tiempo. Esta metodología tiene su origen en 1996 por medio de Kent Beck, Ward Cunningham y Ron Jeffries. (Bahit, 2012)

“Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder; el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar.” Kent Beck

### 1.8.1. Bases de la programación extrema

Son 5 los valores en los que se apoya esta metodología en los cuales se enfatiza el trabajo en equipo, estos valores son los siguientes.

**Comunicación:** En esta metodología es primordial el trabajo en equipo desde el análisis del proyecto hasta la codificación del mismo, procurando llegar a un acuerdo común a los problemas que puedan surgir durante el desarrollo del proyecto.

**Simplicidad:** Lo que busca XP es desarrollar solo lo necesario para así no perder tiempo en puntos que no son requeridos en ese momento. (Bahit, 2012)

**Retroalimentación:** El objetivo de esta metodología es entregar lo necesario en el menor tiempo posible, esto demanda al cliente conocer cómo va el desarrollo actual del proyecto e implementar los cambios que el demande lo más pronto posible, esto es conocido como retroalimentación o feedback-continuo. (Bahit, 2012)

**Respeto:** Existe un respeto mutuo entre el cliente y el equipo a cargo de llevar a cabo el proyecto, esto significa que el equipo respeta la idoneidad del cliente ya que el quien conoce el valor del proyecto, mientras que el cliente respeta la idoneidad del equipo, confiando en ellos en la definición y desarrollo del proyecto. (Bahit, 2012)

**Coraje:** El coraje en esta metodología se refiere a tener el valor de decir la verdad sobre el desarrollo y avance del proyecto, buscando obtener el éxito del mismo y no excusas sobre los errores que puedan surgir. (Bahit, 2012)

### 1.8.2. Principios

La metodología XP cuenta con principios que funcionan como mecanismos que permiten generar practicas concretas a partir de los valores abstractos.

Estos principios están enfocados solamente a la parte del desarrollo de software ayudando aplicar de una mejor manera las prácticas de la metodología, a continuación, se muestran los 14 principios a los que esta apegado la metodología XP (Pantaleo, 2015).

Tabla 7. Principios XP  
Fuente: (Pantaleo, 2015)

Principios	Descripción
<b>Beneficio mutuo</b>	Las actividades realizadas dentro del proyecto deben beneficiar a todos los miembros del equipo.
<b>Diversidad</b>	Las opiniones de todos los miembros del equipo deben ser evaluadas
<b>Humanidad</b>	Necesidades humanas deben ser contempladas.
<b>Responsabilidad aceptada</b>	La responsabilidad no debe ser asignada sino aceptada.
<b>Economía</b>	Las tareas que contemple el proyecto deben sumar valor al negocio.
<b>Calidad</b>	La calidad del proyecto no debe ser utilizada como variable de control
<b>Mejora</b>	Asegurar una mejora continua en los procesos y productos.
<b>Flujo</b>	Entregar de manera continua productos con valor.
<b>Reflexión</b>	Auto interrogarse acerca del trabajo realizado.
<b>Pequeños pasos</b>	Avanzar mediante pequeños incrementos.
<b>Auto semejanza</b>	Reutilizar lo que se ha utilizado anteriormente de manera exitosa.
<b>Falla</b>	Los casos fallidos deben ser utilizados como un mecanismo de aprendizaje.
<b>Oportunidad</b>	Ver a los problemas que surjan durante el proceso de desarrollo como una oportunidad de aprender.



<b>Redundancia</b>	Utilizar más de una alternativa a la vez para resolver problemas.
--------------------	---

### 1.8.3. Prácticas.

Las prácticas son actividades que se llevan a cabo por todos los miembros del proyecto, estas son primarias y corolarias.

Prácticas Primarias: Estas pueden ser aplicadas en cualquier actividad del proyecto además de ser las más recomendadas para empezar. Ver la tabla 8

Prácticas Corolarias: Estas prácticas son recomendables adoptar cuando se haya adquirido experiencia en las practicas primarias. Ver tabla 9

Tabla 8. Practicas primarias metodología XP  
Fuente: (Pantaleo, 2015)

<b>Practicas primarias</b>			
<b>Práctica Primaria</b>	<b>Descripción</b>	<b>Principios que la generan</b>	<b>Valores relacionados</b>
<b>Sentarse Cerca</b>	Disponer de un espacio donde trabaje el equipo completo.	Humanidad, Beneficio Mutuo, Diversidad	Comunicación
<b>Equipo entero</b>	Disponer de personas con habilidades necesarias que aseguren el éxito del proyecto	Diversidad, Economía, Responsabilidad aceptada.	Respeto
<b>Espacio de trabajo informativo</b>	Dentro del espacio del trabajo se debe mostrar el estado del proyecto	Mejora, Reflexión	Comunicación, Retroalimentación
<b>Trabajo energizado</b>	Trabajar cierta cantidad de horas en estado saludable de manera que el trabajo no afecte a la persona ni la productividad	Humanidad	Respeto
<b>Programación de pares</b>	Programar de a dos personas de manera que la funcionalidad asignada sea analizada, diseñada y desarrollada.	Diversidad, Calidad, Reflexión, Beneficio Mutuo	Comunicación, Retroalimentación, Respeto
<b>Historias de usuario</b>	Planificar utilizando historias de usuario: Funcionalidades que son visibles para el cliente y estimadas por el Equipo	Economía, Flujo	Comunicación, Simplicidad
<b>Ciclo semanal</b>	Planificar las actividades semanales en las que se tomen en cuenta los progresos anteriores, el cliente seleccione las	Pequeños incrementos,	Retroalimentación, Comunicación

	historias de la semana y el equipo desglose las historias en tareas	Economía, Reflexión	
<b>Ciclo cuatrimestral</b>	Planificar las actividades cuatrimestrales que estarán alineados a los objetivos de alto nivel	Pequeños incrementos, Economía, Reflexión	Retroalimentación, Comunicación
<b>Slack</b>	Dentro de toda la planificación incluir pequeñas tareas que pueden ser excluidas en caso de necesidad	Flujo	Comunicación
<b>Build de diez minutos</b>	En menos de 10 minutos compilar y ejecutar los tests del sistema	Flujo, Oportunidad, Falla	Retroalimentación
<b>Integración Continua</b>	Si se producen cambios en el sistema estos deben ser integrados y probados en todas partes lo más rápido posible	Flujo, Oportunidad, Pequeños incrementos	Retroalimentación
<b>Desarrollo conducido por las pruebas</b>	Al realizar un cambio, escribir el código que pruebe lo que se modificara	Falla, Pequeños incrementos, Calidad	Coraje, Retroalimentación
<b>Diseño Incremental</b>	Diseñar el sistema de forma gradual cada día	Pequeños incrementos, Calidad	Coraje, Retroalimentación

Tabla 9. Practicas corolarias metodología xp  
Fuente: (Pantaleo, 2015)

#### Practicas Corolarias

<b>Práctica Corolaria</b>	<b>Descripción</b>	<b>Principios que la generan</b>	<b>Valores relacionados</b>
<b>Involucramiento del cliente</b>	Toda persona que sea afectada por el sistema tiene que ser parte del equipo	Flujo, Economía	Comunicación, Retroalimentación
<b>Entregas Incrementales</b>	Si se reemplaza un sistema existente, hacerlo por partes	Flujo, pequeños pasos	Retroalimentación, Coraje, Simplicidad
<b>Continuidad del equipo</b>	Los equipos efectivos deberán mantenerse juntos	Humanidad	Comunicación, Simplicidad
<b>Contraer equipos</b>	Mantener una carga de trabajo constante de acuerdo con la productividad y número de personas en los equipos	Mejora	Simplicidad
<b>Análisis de causas</b>	Eliminar los problemas y sus respectivas causas	Mejora	Simplicidad
<b>Código Compartido</b>	El sistema puede ser mejorado por cualquier miembro del equipo a cualquier momento	Reflexión, mejora	Retroalimentación
<b>Código y pruebas</b>	Siempre se realizarán pruebas y se analizara el código para que después se pueda extraer lo necesario a partir de ellos	Economía, Calidad	Retroalimentación
<b>Línea Base única del repositorio del código</b>	Siempre intentar desarrollar una única línea de desarrollo, salvo que sea estrictamente necesario abrir otra rama,	Flujo	Simplicidad, Comunicación

	pero hacerlo por el menor tiempo posible		
<b>Instalación diaria</b>	Todos los días se instalarán nuevas versiones del sistema	Flujo, Pequeños pasos, Economía, Calidad	Retroalimentación, Coraje, Simplicidad
<b>Contrato de alcance negociable</b>	Mientras el proyecto siga en desarrollo establecer contratos que propicien una negociación de alcance	Flujo, Economía	Coraje, Simplicidad
<b>Pagar por usar</b>	Intentar cobrar por el uso del sistema en conjunto con el desarrollo del software	Economía	Retroalimentación

## CAPÍTULO II. DESARROLLO

### 2.1. Planificación del proyecto

Dentro de la metodología XP, la planificación del proyecto nos permite conocer las necesidades del cliente, establecer los requerimientos reales y crear y entregar un producto final de calidad.

En la figura 7, se muestra el patrón que utiliza la metodología XP con la descripción que debe cumplir en cada ciclo.

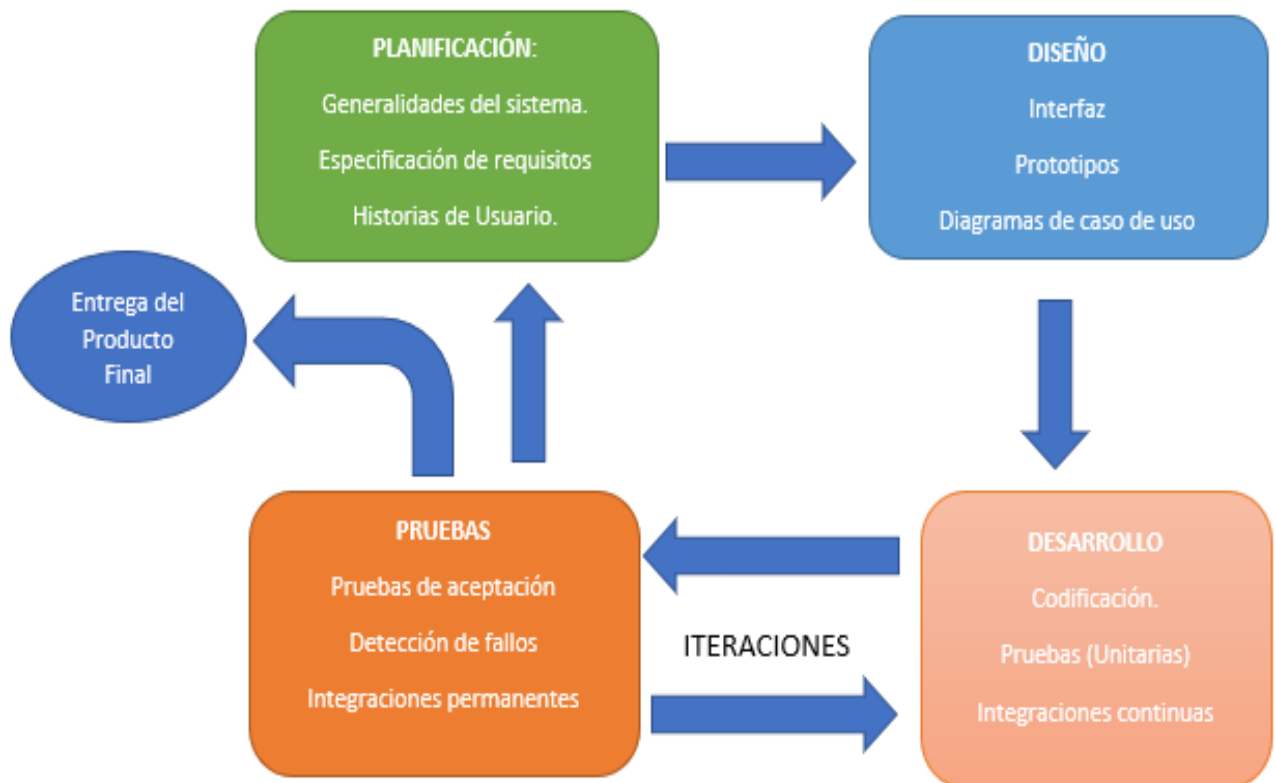


Figura 7. Metodología XP ciclo de vida

### 2.1.1. Proceso de gestión de planificaciones curriculares

El diagrama descrito a continuación en la figura 8, especifica todos los pasos que cumple la institución para llevar a cabo el proceso de gestión de planificaciones curriculares el cual fue descrito a detalle en el capítulo 1.3.

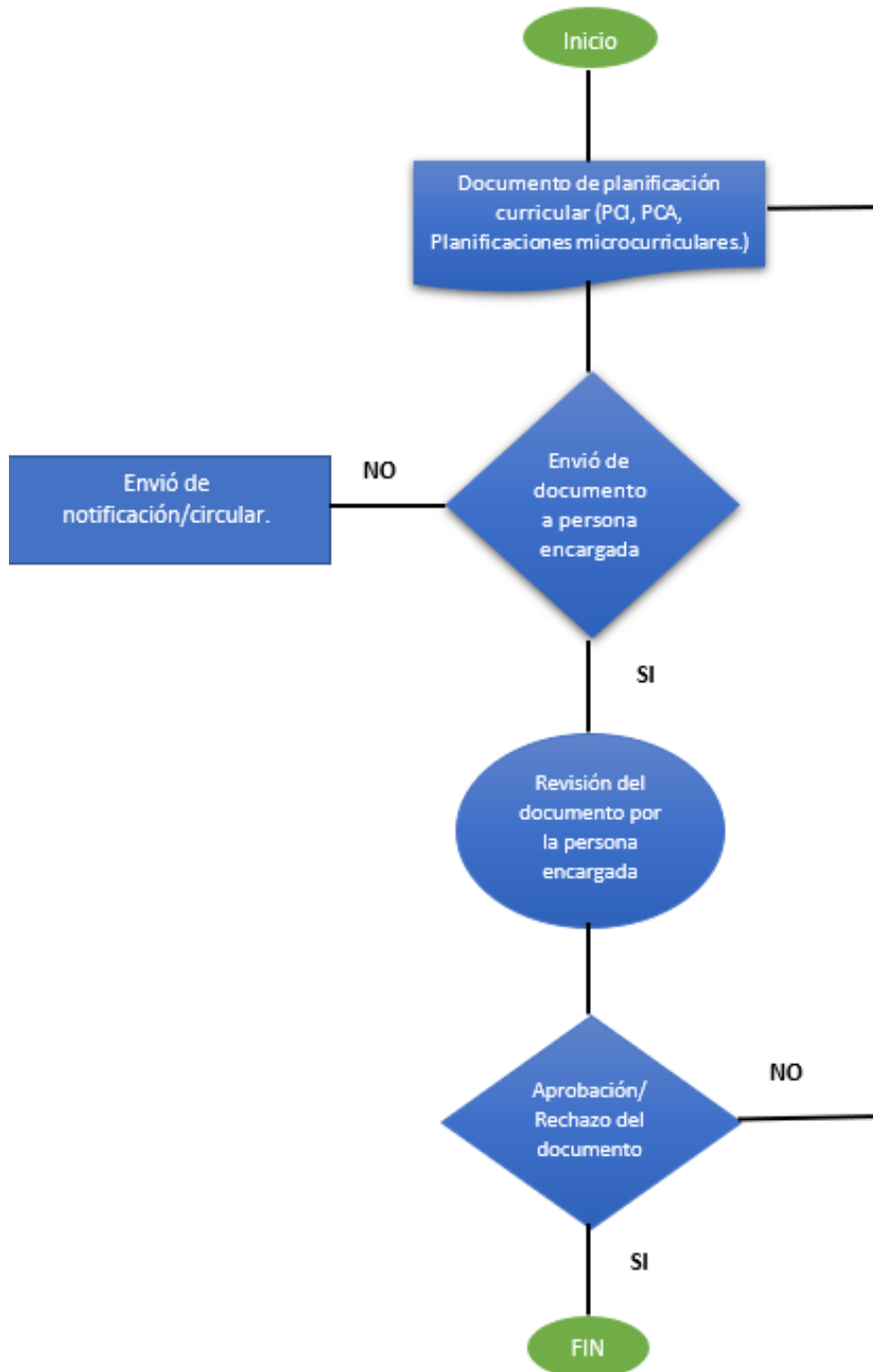


Figura 8. Diagrama de flujo del proceso de planificación curricular

## 2.1.2. Especificación de requerimientos

De acuerdo con las reuniones mantenidas con el rector y la persona que administrara el sistema, se han llegado a establecer los siguientes requerimientos.

## 2.1.3. Tipos de Usuario

- **Usuario Administrador.** Permite el control del módulo administrativo del sistema.
- **Usuario Rector.** Se encargará de validar las planificaciones subidas al portal, rechazarlas y analizarlas; además de subir las PCI podrá enviar circulares a los docentes.
- **Usuario Jefe de Área.** Es el encargado de subir las PCA y enviar circulares a los docentes encargados de las asignaturas del área.
- **Usuario Docente.** Es el encargado de subir al portal las Planificaciones de Aula y podrá revisar los circulare.

## 2.1.4. Historias de Usuario

Durante el desarrollo de la metodología XP existe una manera de expresar un requisito de una manera simple y en términos del usuario, por lo tanto se puede expresar una necesidad del usuario de una forma no formal, esto es conocido como user story (Historias de Usuario) y bien pueden ser representadas como una simple oración, como el siguiente ejemplo. (Fontela, 2013)

Como <rol>

Quiero que el sistema haga <Funcionalidad>

Para obtener <beneficio esperado>

Las historias de usuario generadas de las distintas reuniones con los encargados de la UE son:

Tabla 10. Historia de usuario 01

<b>Historia de Usuario</b>	
<b>Número: 01</b>	<b>Usuario: Administrador – Rector – Jefe de Área - Docentes</b>
<b>Nombre historia: Acceso al Portal</b>	

<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador:</b> Sr. Anthony Valverde	<b>Iteración asignada:</b> 1
<b>Descripción:</b> Los usuarios puedan ingresar al portal desde cualquier navegador y dispositivo a cualquier hora del día, cualquier día de la semana.	

Tabla 11. Historia de usuario 02

<b>Historia de Usuario</b>	
<b>Número:</b> 02	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Gestión del portal.	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Bajo
<b>Programador:</b> Sr. Anthony Valverde	<b>Iteración asignada:</b> 2
<b>Descripción:</b> El administrador podrá gestionar usuarios, docentes, áreas del conocimiento, materias, cursos, periodos. No podrá autorizar las planificaciones solo podrá visualizarlas. No podrá enviar circulares.	

Tabla 12. Historia de usuario 03

<b>Historia de Usuario</b>	
<b>Número:</b> 03	<b>Usuario:</b> Rector
<b>Nombre historia:</b> Gestión de Circulares y Planificaciones.	
<b>Prioridad:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Programador:</b> Sr. Anthony Valverde	<b>Iteración asignada:</b> 3
<b>Descripción:</b> El rector podrá visualizar las planificaciones subidas por los docentes y jefes de área, aprobar o rechazar una planificación enviando la respectiva observación. Enviar circulares ya sea a docentes o jefes de área.	

Tabla 13. Historia de usuario 04

<b>Historia de Usuario</b>	
<b>Número:</b> 04	<b>Usuario:</b> Jefe de Área
<b>Nombre historia:</b> Publicar los PCA y Enviar Circulares.	
<b>Prioridad:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Programador:</b> Sr. Anthony Valverde	<b>Iteración asignada:</b> 4
<b>Descripción:</b> El usuario Jefe de Área podrá publicar los PCA para que sean aprobados por el usuario Rector, además, tendrá la capacidad de enviar circulares a los Docentes de las asignaturas dentro del Área del Conocimiento.	

Tabla 14. Historia de usuario 05

<b>Historia de Usuario</b>	
<b>Número:</b> 05	<b>Usuarios:</b> Docente
<b>Nombre historia:</b> Publicar Planificación de Aula y Enviar Circulares.	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Bajo
<b>Programador:</b> Sr. Anthony Valverde	<b>Iteración asignada:</b> 4
<b>Descripción:</b> El usuario docente podrá publicar las Planificaciones de Aula para que sean aprobadas o rechazadas por el Rector, además tendrá la capacidad de enviar circulares en caso de que sea necesario.	

Tabla 15. Historia de usuario 06

<b>Historia de Usuario</b>	
<b>Número:</b> 06	<b>Usuario:</b> Rector – Jefe de Área - Docente
<b>Nombre historia:</b> Buzón de Circulares	
<b>Prioridad:</b> Baja	<b>Riesgo en desarrollo:</b> Bajo
<b>Programador:</b> Sr. Anthony Valverde	<b>Iteración asignada:</b> 5
<b>Descripción:</b> Todos los usuarios podrán acceder al respectivo buzón de circulares que les hayan sido enviadas.	

Adicionalmente se requiere de ciertas funcionalidades específicas como:

- Seguridad
- Disponibilidad
- Adaptabilidad en Dispositivos
- Manejos de Esquemas de seguridad
- Manejo de herramientas libres para el desarrollo de la aplicación.

### 2.1.5. Definición de acrónimos y Abreviaturas

PCI. Planificación Curricular Institucional

PCA. Planificación Curricular Anual (Planificación por Área de conocimiento)



## 2.2. Descripción General

### 2.2.1. Perspectiva del Producto

La aplicación será un Portal Web y una aplicación Móvil. Respecto al portal web, este permitirá gestionar todo el proceso de planificación curricular en la que cada persona tendrá acceso a diferentes funciones del sistema de acuerdo al rol y cargo que tenga además de tener acceso a un buzón de circulares, mientras que la aplicación móvil será un acceso rápido al buzón de circulares.

### 2.2.2. Funcionalidades del Proyecto

La aplicación es de tipo administrador diferenciada por las diferentes características de los perfiles de usuarios.

Tabla 16. Funcionalidades del proyecto

Administrador	Seguridad	Personal
		Usuarios
	Periodo	Materias
		Cursos
		Paralelos
		Planificación
Rector	Planificaciones	PCI
		PCA
		Planificaciones de Aula
	Circulares	Crear Circular
		Buzón
Usuario	Jefe de Área	Subir PCA
		Crear Circulares
	Docente	Planificaciones de Aula
	Circulares	Buzón

Tabla 17. Funcionalidades del proyecto-Administrador

Tipo de Usuario	Administrador
Formación	Ingeniero en Sistemas
Habilidades	Manejo de sistemas operativos, administración de base de datos, administración de portales web

<b>Actividades</b>	Control Total del Sistema
--------------------	---------------------------

Tabla 18. Funcionalidades del proyecto-Rector

<b>Tipo de Usuario</b>	<b>Rector</b>
<b>Formación</b>	Docente
<b>Habilidades</b>	Manejo de Portales Web
<b>Actividades</b>	Autorizar Planificaciones. Envió de Circulares

Tabla 19. Funcionalidades del proyecto-Jefe de Área

<b>Tipo de Usuario</b>	<b>Jefe de Área</b>
<b>Formación</b>	Docente
<b>Habilidades</b>	Manejo de Portales Web
<b>Actividades</b>	Envió de Planificaciones. Envió de Circulares

Tabla 20. Funcionalidades del proyecto-Docentes

<b>Tipo de Usuario</b>	<b>Docentes</b>
<b>Formación</b>	Docente
<b>Habilidades</b>	Manejo de Portales Web
<b>Actividades</b>	Autorizar Planificaciones. Envió de Circulares

### 2.2.3. Restricciones de la Aplicación

La conectividad y las herramientas de desarrollo definen las restricciones de la aplicación.

- Herramientas Libres
- Acceso a través del internet

## 2.2.4. Requisitos de Interfaz

Los requisitos de interfaz son especificaciones por parte del usuario en las que plasma una idea tanto de la funcionalidad y del contenido que quiere ver en el producto final. Estos pueden ser explicados de una manera no formal y pueden ser mejorados por sugerencias del programador responsable del diseño del sistema. (Wood, 2015)

A continuación, se muestra los requisitos de interfaz especificados por el cliente.

La aplicación dispone de dos interfaces, la interfaz web (Administración, Usuario) y una aplicación móvil para las circulares.

La interfaz web de administrador es la zona donde se gestionan los parámetros de funcionalidad de sistema y se lleva a cabo el proceso de gestión curricular.

La interfaz web de usuario será la zona en la que se realizara él envío de documentos y revisión de circulares.

## 2.2.5. Requisitos de Software

Las herramientas de desarrollo son de la aplicación web son de libre distribución.

- Framework CodeIgniter.
- Framework Ionic versión 2

## 2.2.6. Requisitos Funcionales

Los requisitos funcionales derivados del análisis de las historias de usuarios son:

Tabla 21. Requisitos Funcionales

<b>Número de requisito</b>	<b>RF-01</b>
<b>Nombre de requisito</b>	Gestión de Personal
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
<b>Fuente del requisito</b>	Crea, editar, eliminar personas
	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<b>Número de requisito</b>	<b>RF-02</b>
<b>Nombre de requisito</b>	Gestión de Usuario
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
<b>Fuente del requisito</b>	Creación de usuarios
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<b>Número de requisito</b>	<b>RF-03</b>
<b>Nombre de requisito</b>	Asignación de rector

<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
<b>Fuente del requisito</b>	Asignar al usuario Rector.
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<b>Número de requisito</b>	<b>RF-04</b>
<b>Nombre de requisito</b>	Gestión de Planificaciones
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
<b>Fuente del requisito</b>	Autorizar o Rechazar las Planificaciones.
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<b>Número de requisito</b>	<b>RF-05</b>
<b>Nombre de requisito</b>	Gestión de Circulares
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
<b>Fuente del requisito</b>	Envío de circulares a docentes y jefes de área.
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

### 2.2.7. Requisitos No funcionales

Los requisitos no funcionales derivados del análisis son:

Tabla 22.Requisitos No Funcionales

<b>Número de requisito</b>	<b>RNF-01</b>
<b>Nombre de requisito</b>	Interfaz adaptativa
<b>Descripción</b>	Cumplir los estándares que permitan desplegar la aplicación en cualquier tipo de navegador.
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<b>Número de requisito</b>	<b>RNF-02</b>
<b>Nombre de requisito</b>	Disponibilidad del sistema
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito
<b>Descripción</b>	Deberá estar disponible 24 horas al día, 7 días a la semana.
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<b>Número de requisito</b>	<b>RNF-03</b>
<b>Nombre de requisito</b>	Seguridad de la Información
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito
<b>Descripción</b>	Garantizar a los usuarios la seguridad en la información.
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<b>Número de requisito</b>	<b>RNF-04</b>
<b>Nombre de requisito</b>	Arquitectura basada en WAMP
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito

<b>Descripción</b>	La arquitectura que maneje el sistema tiene que estar basada en las siguientes herramientas Windows, Mysql y php como lenguaje de programación.
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

## 2.3. Implementación del prototipo

El prototipo permite validar los requerimientos de la aplicación, tomando en cuenta las historias de los usuarios.

### 2.3.1. Especificación de los casos de uso

Un caso de uso es una secuencia de tareas que son desarrolladas por un sistema en respuesta a un evento que inicia un actor. Los diagramas de casos de uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los actores. Tanto un caso de uso como los diagramas de caso no son destacados en las metodologías ágiles, pero su uso es aceptable ya que muestra de una forma más formal y gráfico la interpretación de un requisito. (Fontela, 2013).

En las siguientes figuras se mostrarán todos los casos de uso que tomaran en cuenta para el sistema de gestión de planificaciones curriculares.

Actores del sistema.

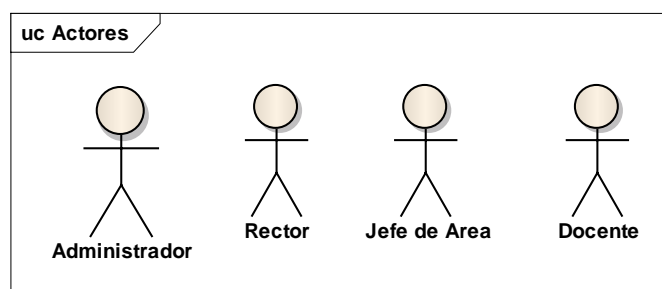


Figura 9. Casos de uso-Actores

## Actor administrador y sus funciones.

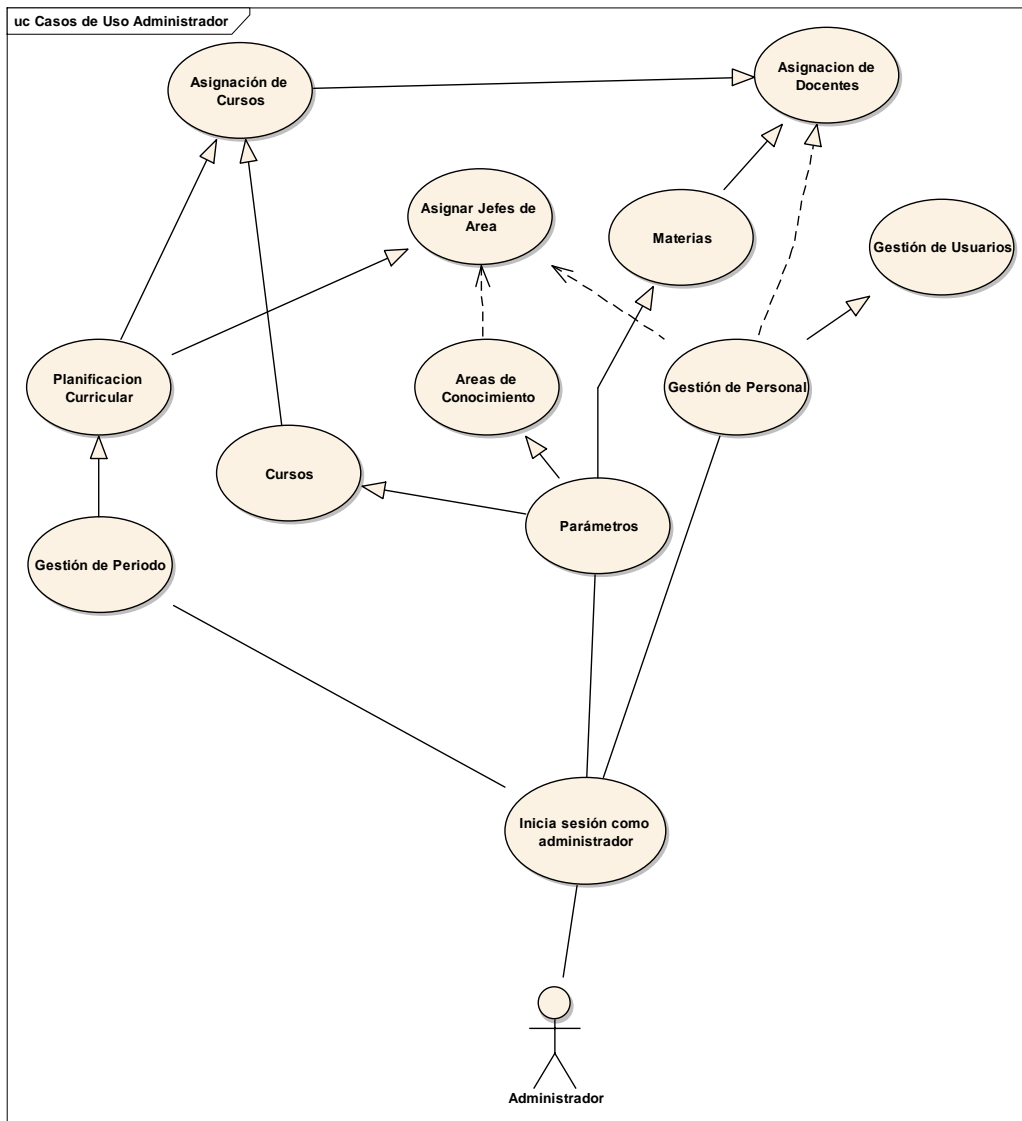


Figura 10. Casos de uso-Administrador

### Actor Rector y sus funciones.

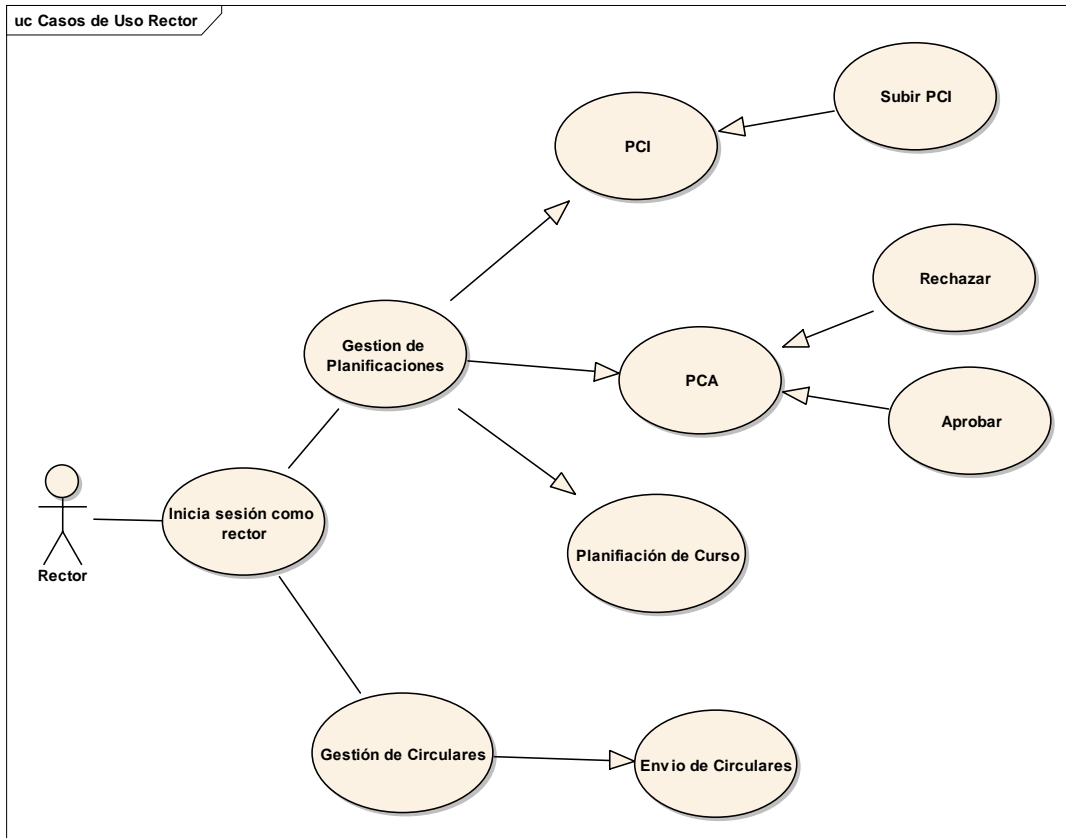


Figura 11. Casos de uso-Rector

### Actor Jefe de Área y sus funciones.

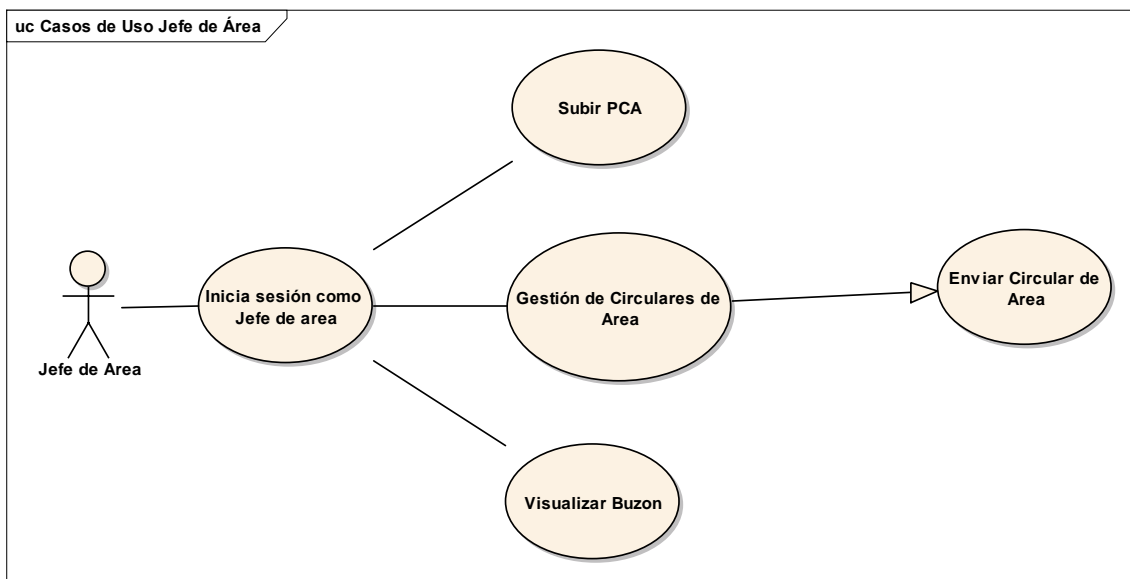


Figura 12. Casos de uso-Jefe de Área

Actor Docente y sus funciones.

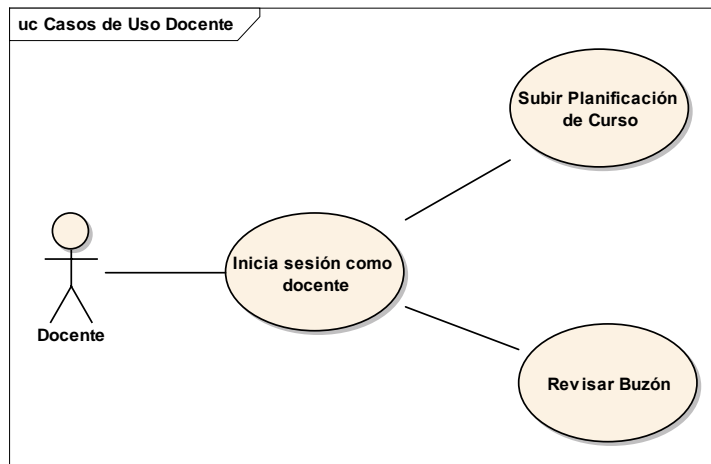


Figura 13. Casos de uso-Docente

### 2.3.2. Casos de Uso de la aplicación

Tabla 23. Descripción de caso de uso

<b>Código: CU-01</b>	<b>Nombre del Caso de Uso: Ingreso al portal</b>	
<b>Objetivo</b>	Controlar el ingreso al portal dependiendo del rol de usuario	
<b>Descripción</b>	Al ingresar al portal el sistema filtrara información y funcionalidades dependiendo del tipo de rol de usuario	
<b>Precondición</b>	Se debe tener un dispositivo con acceso a internet y a un navegador web. Tener la información del usuario dentro de la base de datos	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario debe encontrarse en la página de ingreso del sistema
	2	Insertara su usuario y contraseña
	3	El sistema mostrara la interfaz de acuerdo con el tipo de usuario ingresado
	4	Fin del caso de uso
<b>Postcondición</b>	Las funcionalidades a las que pueda acceder el usuario se verán reflejadas en la interfaz del sistema	
<b>Frecuencia esperada</b>	200	
<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-02</b>	<b>Nombre del Caso de Uso: Gestión de periodos.</b>	



<b>Objetivo</b>	Administrar los periodos que se encuentra vigentes la institución.
<b>Descripción</b>	La persona debe tener el rol de administrador dentro del sistema El sistema permitirá al usuario con rol de administrador acceder a la gestión de periodos.
<b>Precondición</b>	Tener una cuenta de usuario con rol de administrador
<b>Secuencia</b>	<b>Paso    Acción</b>
<b>Normal</b>	1        EL usuario ingresa al sistema como administrador.
	2        El sistema despliega la interfaz para el rol de Administrador
	3        En el menú dirigirse a la sección de periodos
	4        Administrar los periodos
	5        Fin del caso de uso
<b>Postcondición</b>	Quedaran registrados los cambios en la sección de periodos
<b>Frecuencia esperada</b>	50
<b>Comentarios</b>	Sin Comentarios
<b>Código: CU-03</b>	<b>Nombre del Caso de Uso: Gestión de usuarios.</b>
<b>Objetivo</b>	Administrar los usuarios del sistema
<b>Descripción</b>	El sistema permitirá al usuario con rol de administrador acceder a la gestión de usuarios.
<b>Precondición</b>	La persona debe tener el rol de administrador dentro del sistema
<b>Secuencia</b>	<b>Paso    Acción</b>
<b>Normal</b>	1        El usuario accede al sistema como administrador
	2        El sistema despliega la interfaz para el rol de Administrador
	3        En el menú dirigirse a la sección de usuarios
	4        Administrar los usuarios
	5        Fin del caso de uso
<b>Postcondición</b>	Quedaran registrados los cambios de información de usuarios dentro de la base de datos
<b>Frecuencia esperada</b>	100
<b>Comentarios</b>	Sin Comentarios
<b>Código: CU-04</b>	<b>Nombre del Caso de Uso: Gestión de áreas de conocimiento</b>

<b>Objetivo</b>	Administrar las áreas de conocimiento	
<b>Descripción</b>	El sistema permitirá al usuario con rol de administrador acceder a la gestión de áreas de conocimiento crear modificar o borrar áreas de conocimientos.	
<b>Precondición</b>	La persona debe tener el rol de administrador dentro del sistema	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema como administrador
	2	El sistema despliega la interfaz para el rol de Administrador
	3	En el menú dirigirse a la sección de áreas de conocimiento
	4	Administrar las áreas de conocimiento
	5	Fin del caso de uso
<b>Postcondición</b>	Quedaran registrados los cambios de áreas de conocimiento dentro de la base de datos	
<b>Frecuencia esperada</b>	100	
<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-05</b>	<b>Nombre del Caso de Uso: Gestión de materias</b>	
<b>Objetivo</b>	Administrar las materias que se encuentran vigentes en la institución	
<b>Descripción</b>	El sistema permitirá al usuario con rol de administrador acceder a la gestión de materias en donde podrán crear modificar o borrar áreas de conocimientos.	
<b>Precondición</b>	La persona debe tener el rol de administrador dentro del sistema	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema como administrador
	2	El sistema despliega la interfaz para el rol de Administrador
	3	En el menú dirigirse a la sección de materias
	4	Administrar las materias como sea necesario
	5	Fin del caso de uso
<b>Postcondición</b>	Quedaran registrados los cambios de materias dentro de la base de datos	
<b>Frecuencia esperada</b>	100	

<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-06</b>	<b>Nombre del Caso de Uso: Gestión de cursos</b>	
<b>Objetivo</b>	Administrar los cursos que se encuentran vigentes en la institución	
<b>Descripción</b>	La persona debe tener el rol de administrador dentro del sistema	
<b>Precondición</b>	Tener una cuenta de usuario con rol de administrador	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema como administrador
	2	El sistema despliega la interfaz para el rol de Administrador
	3	En el menú dirigirse a la sección de cursos
	4	Administrar los cursos
	5	Fin del caso de uso
<b>Postcondición</b>	Quedaran registrados los cambios de cursos dentro de la base de datos	
<b>Frecuencia esperada</b>	100	
<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-07</b>	<b>Nombre del Caso de Uso: Gestión de Planificaciones</b>	
<b>Objetivo</b>	Administrar las diferentes planificaciones subidas al sistema	
<b>Descripción</b>	El sistema permitirá al usuario con rol Rector acceder a la gestión de planificaciones (institucional, de área y de curso).	
<b>Precondición</b>	Tener una cuenta de usuario con rol Jefe de Área	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema como Rector
	2	El sistema despliega la interfaz para el rol de Rector
	3	En el menú dirigirse a la sección de planificaciones
	4	Seleccionar entre las opciones la requerida por el usuario
	5	Fin del caso de uso
<b>Postcondición</b>	La planificación institucional quedara guardada en el sistema.	

	Cambios dentro de la gestión de planificaciones quedaran guardados además de mostrarse los cambios pertinentes a los usuarios implicados.	
<b>Frecuencia esperada</b>	400	
<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-08</b>	<b>Nombre del Caso de Uso: Gestión de circulares</b>	
<b>Objetivo</b>	Administrar el envío de circulares	
<b>Descripción</b>	El sistema permitirá al usuario acceder a la sección de circulares para enviar notificaciones a las personas correspondientes	
<b>Precondición</b>	Tener una cuenta de usuario dentro del sistema	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema
	2	El sistema despliega la interfaz de acuerdo al tipo de usuario
	3	En el menú dirigirse a la sección de circulares
	4	Enviar una circular a un usuario requerido
	5	Fin del caso de uso
<b>Postcondición</b>	La información de la circular será guardada en la base de datos. El usuario implicado podrá revisar el circular.	
<b>Frecuencia esperada</b>	100	
<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-09</b>	<b>Nombre del Caso de Uso: Visualizar envíos de planificaciones</b>	
<b>Objetivo</b>	Revisar las planificaciones que han sido enviadas	
<b>Descripción</b>	El sistema permitirá al usuario con rol Rector acceder a la gestión de planificaciones revisarlas aceptarlas o rechazarlas.	
<b>Precondición</b>	La persona debe estar registrada como Rector en la base de datos	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema como Rector
	2	El sistema despliega la interfaz para el rol de Rector

	3	En el menú dirigirse a la sección de planificación curricular o jefe de área.
	4	Revisar nuevos envíos de documentos
	5	Aceptar o Rechazar el documento
	6	Fin del caso de uso
<b>Postcondición</b>	En caso de ser aceptado o rechazado el documento, la persona implicada podrá ver la notificación correspondiente.	
<b>Frecuencia esperada</b>	200	
<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-10</b>	<b>Nombre del Caso de Uso: Subir PCA</b>	
<b>Objetivo</b>	Subir la planificación curricular de área	
<b>Descripción</b>	El sistema permitirá al usuario designado como Jefe de Área subir la planificación curricular de área para su respectiva revisión.	
<b>Precondición</b>	La persona debe estar registrada dentro de la tabla de personal en la base de datos y ser asignado como jefe de área de un área de conocimiento	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema como Jefe de Área
	2	El sistema despliega la interfaz para el rol de Jefe de Área
	3	En el menú dirigirse a la sección de PCA
	4	Subir el documento de planificación.
	5	Fin del caso de uso
<b>Postcondición</b>	El documento quedará guardado en el sistema y podrá ser revisado por la persona correspondiente	
<b>Frecuencia esperada</b>	200	
<b>Comentarios</b>	Sin Comentarios	
<b>Código: CU-11</b>	<b>Nombre del Caso de Uso: Planificaciones de curso.</b>	
<b>Objetivo</b>	Subir planificaciones de curso	
<b>Descripción</b>	El sistema permitirá al usuario con rol docente subir las planificaciones de aula al sistema.	
<b>Precondición</b>	Tener una cuenta de usuario con rol docente	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema como Docente
	2	El sistema despliega la interfaz para el Docente

	3	En el menú dirigirse a la sección de Planificaciones de curso.
	4	Subir el documento de planificación de aula.
	5	Fin del caso de uso
<b>Postcondición</b>	El documento quedará guardado en el sistema y podrá ser revisado por la persona correspondiente	
<b>Frecuencia esperada</b>	200	
<b>Comentarios</b>	Sin comentarios	
<b>Código: CU-12</b>	<b>Nombre del Caso de Uso: Visualizar Buzón.</b>	
<b>Objetivo</b>	Revisar circulares	
<b>Descripción</b>	El sistema permitirá a los usuarios entrar a su buzón de circulares y revisar mensajes enviados y recibidos.	
<b>Precondición</b>	Tener un usuario registrado dentro de la base de datos	
<b>Secuencia</b>	<b>Paso</b>	<b>Acción</b>
<b>Normal</b>	1	El usuario accede al sistema con su usuario
	2	El sistema despliega la interfaz para el rol de usuario correspondiente
	3	En el menú dirigirse a la sección de Buzón.
	4	Revisar los circulares.
	5	Fin del caso de uso
<b>Postcondición</b>	El usuario podrá enviar una respuesta si es necesaria.	
<b>Frecuencia esperada</b>	200	
<b>Comentarios</b>	Sin Comentarios.	

## 2.4. Diseño de la Aplicación

### 2.4.1. Arquitectura del sistema

La arquitectura del sistema es una arquitectura de despliegue que está basada en el patrón MVC (Modelo Vista Controlador) el cual ya está aplicado en el framework CodeIgniter, este nos permite una manera más organizada de programar, facilitando posibles cambio e integraciones de nuevas funciones a futuro.

Primero tenemos a la capa modelo, esta capa es la que se encarga de establecer la comunicación con la base de datos para acceder a la información del sistema, la base de datos con la que trabaja CodeIgniter es MySQL por eso dentro de sus paquetes viene incluido el driver que permita la comunicación entre la base de datos y el sistema.

Después esta la capa de controlador, esta capa es la que establece la comunicación entre las otras 2 capas siendo esta en donde se establece la lógica del negocio/sistema recibiendo peticiones de los usuarios para realizar el pedido a la capa modelo y devolver la información a la capa vista, su codificación se encuentra en lenguaje de programación Php con un modo de uso propio del Framework.

La capa vista es la encargada de mostrar la información que recibe como respuesta de la capa de controlador ante un pedido de usuario esta realizada a base de código html, css y javascript; mostrando como resultado una vista de la información agradable para el usuario.

Por último, se encuentra el despliegue, que es la forma en la que se muestra el sistema al usuario, esta es una página web estructurada de manera amigable para el usuario capaz de funcionar dentro de un navegador.

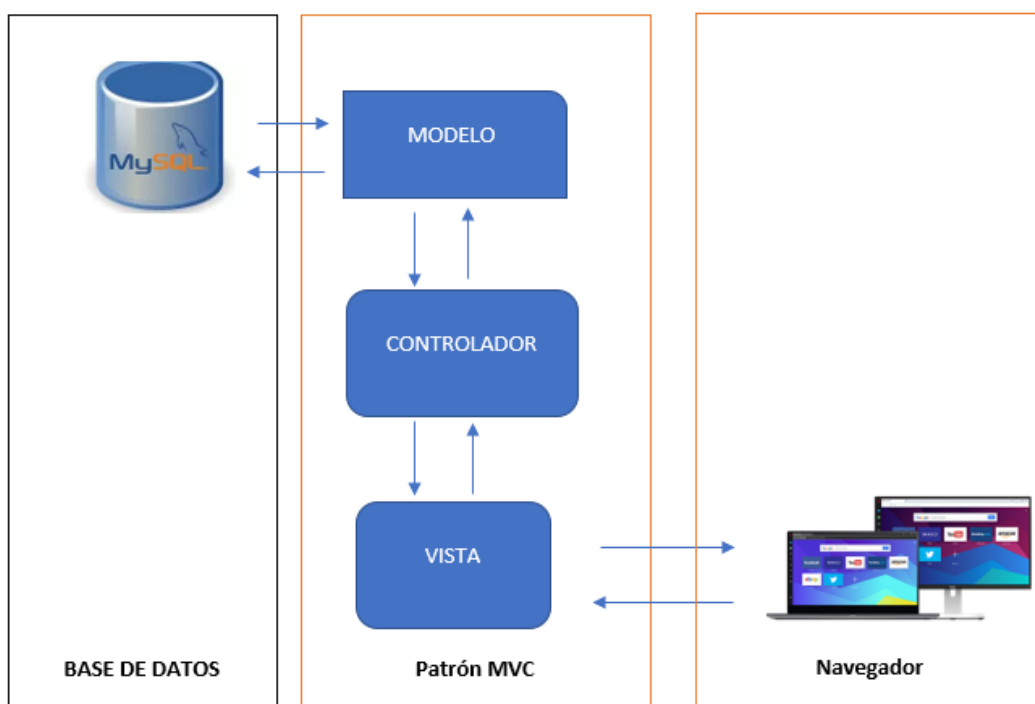


Figura 14. Arquitectura de Despliegue

## 2.4.2. Diagrama de Base de Datos

En la figura 15, se muestra el diagrama de entidad relacion de la base de datos, con la respectiva descripción de campos que contiene cada tabla. Este diagrama servirá como base para el desarrollo de la aplicación.

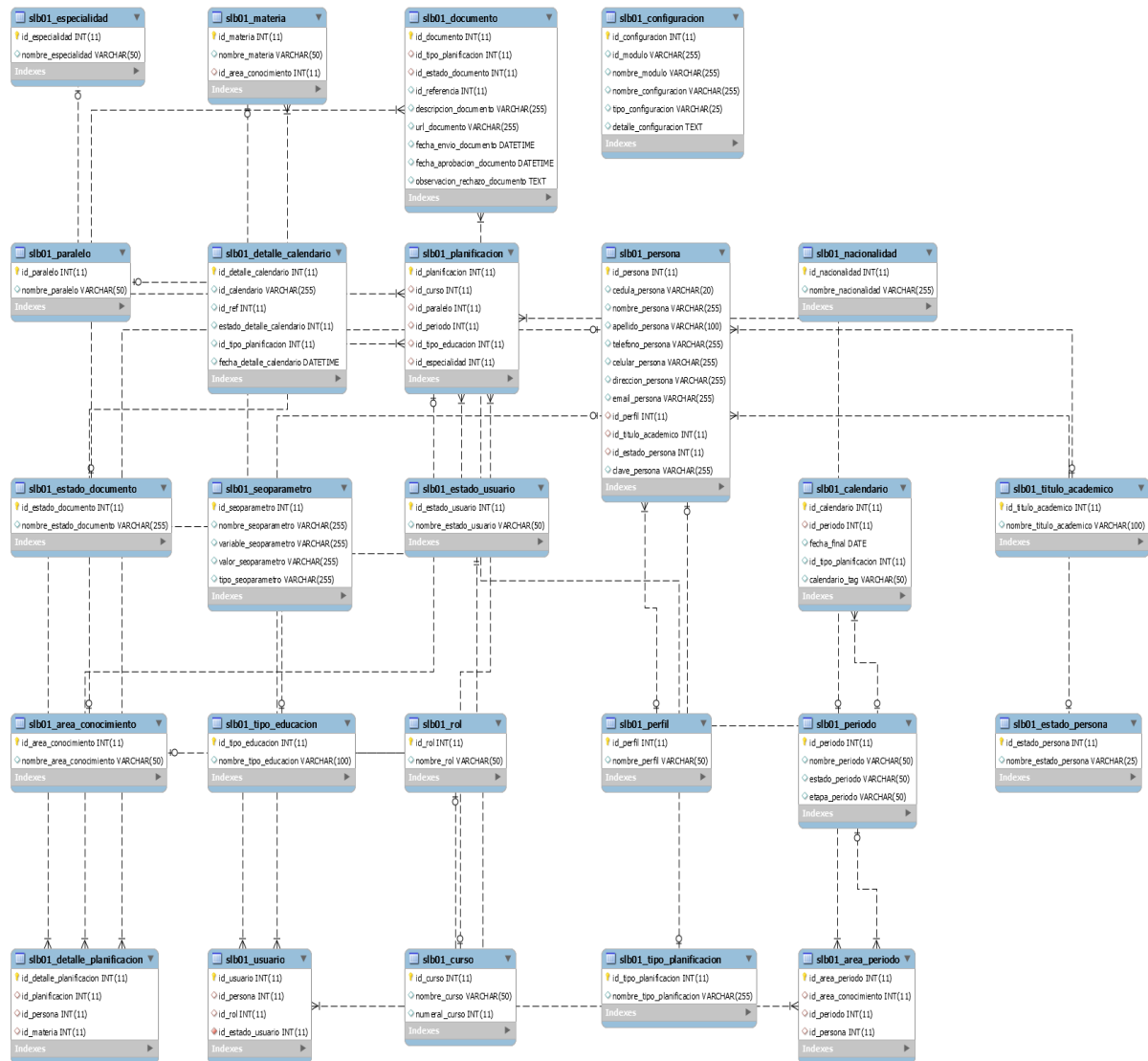


Figura 15. Diagrama de la Base de Datos



### 2.4.3. Diagrama de Paquetes

El diagrama de paquetes que se ve en la figura 16 nos muestra la distribución de paquetes del sistema, tal distribución ya viene dada por el framework, lo que ayuda a desarrollar un sistema organizado.

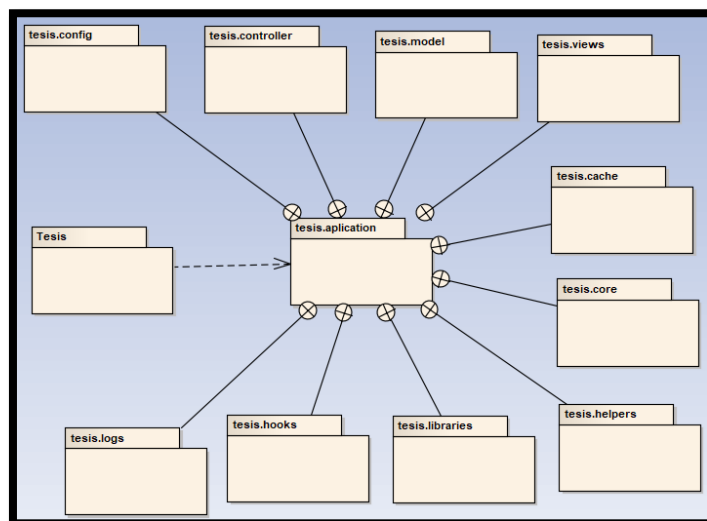


Figura 16. Diagrama de paquetes

## 2.5. Iteraciones

De acuerdo a la metodología XP, para cumplir con las expectativas del sistema existen iteraciones, estas son reuniones con el cliente en las que se muestran avances del sistema cada cierto tiempo, aquí se plasma el cumplimiento de lo acordado en las historias de usuario, aquí se corrigen fallas o se modifican ciertas partes del sistema que el cliente requiera, cambios que no se salen de lo establecido en un principio, con esto se asegura que el sistema cumpla con las especificaciones del cliente.

## 2.6. Primera Iteración

Se desarrollo ya la interfaz que usara el sistema y se agregó funcionalidades básicas tales con ingresos de información a la base de datos.

### 2.6.1. Historia de Usuario 1

De acuerdo con lo establecido en la historia de usuario 1 (véase tabla 10), se pide establecer un ingreso a portal dentro de cualquier navegador y dispositivo para cualquier tipo de usuario.

- **Ingreso al portal**

Para ingresar al portal, es necesario tener una cuenta de usuario (usuario y contraseña). El sistema estará diferenciado por 2 interfaces diferente una para la parte administrativa y otra para la parte del personal.

Para la parte administrativa utilizará el siguiente enlace.

[planificacionspp.com/index.php/administrator](https://planificacionspp.com/index.php/administrator).

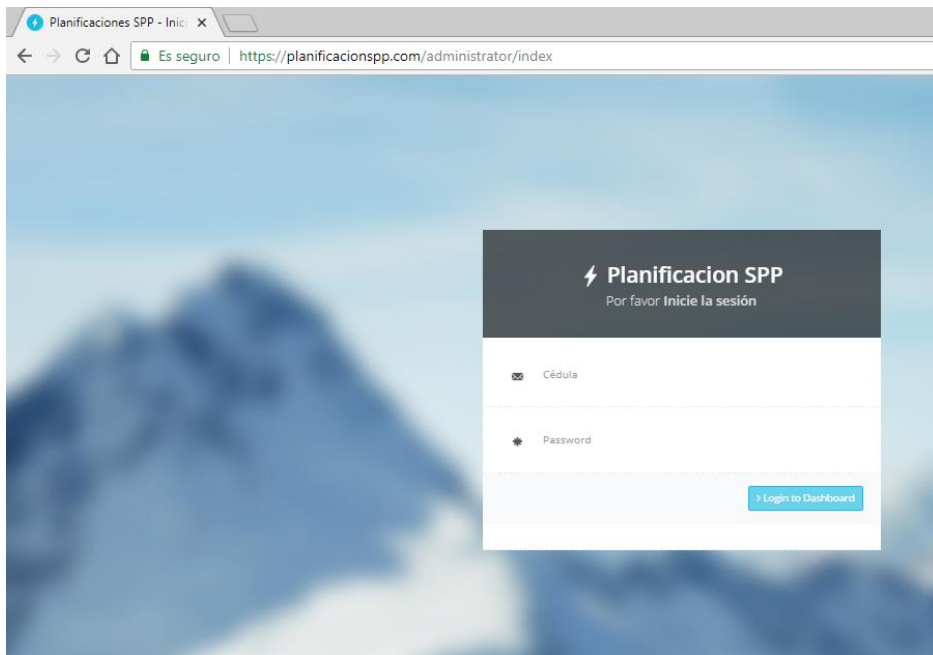


Figura 17. Ingreso portal administrador

Al ingresar al sistema se encontrará con la siguiente interfaz

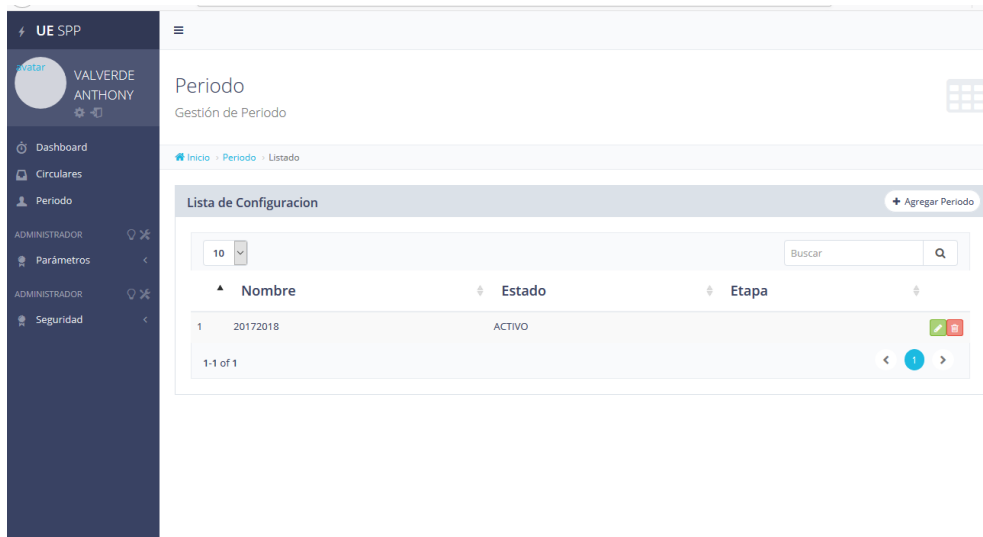


Figura 18. Administrador Interfaz

Para la parte de personal esta ingresara con el siguiente enlace [planificacionspp.com/index.php/personalfrent/index](https://planificacionspp.com/index.php/personalfrent/index)

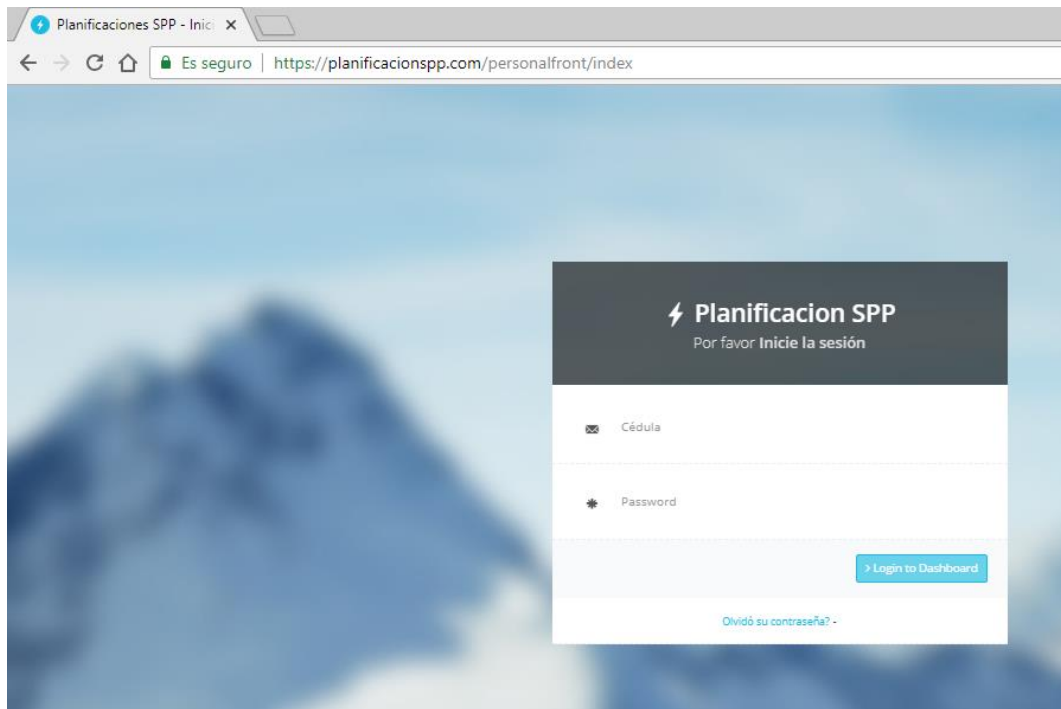


Figura 19. Ingreso porta personal

Al ingresar al sistema se encontrará con la siguiente interfaz

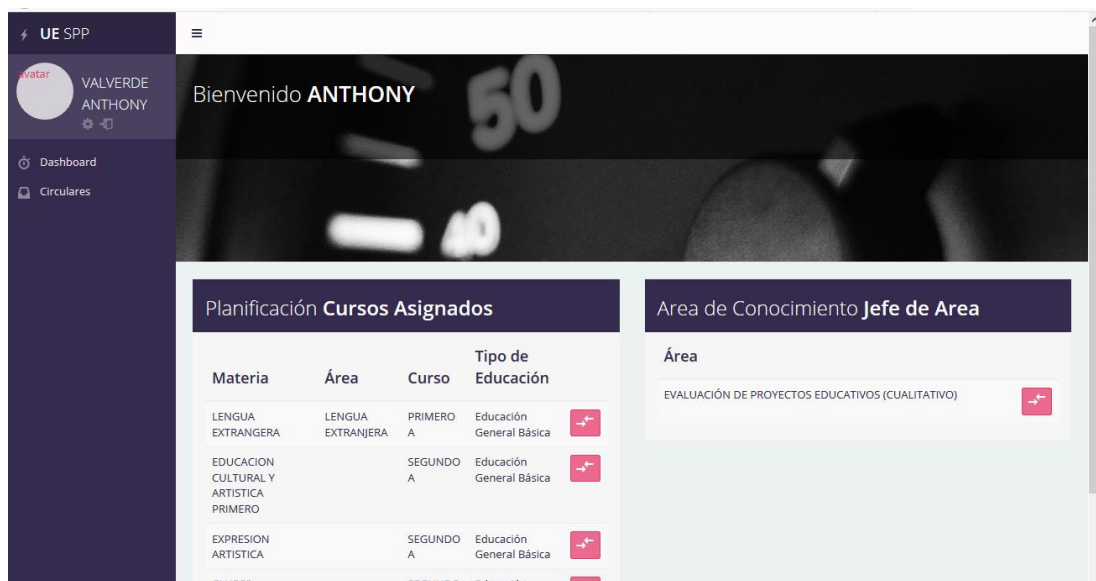


Figura 20. Interfaz de personal

- Validación de usuarios.

La validación usuarios está dada de manera que existen 3 roles (Administrador, Rector, Docente). Solo los usuarios que son asignados como administrador o rector podrán ingresar desde la parte de administrador, pero todas las personas podrán ingresar a la parte de personal ya que existen personas que puedan tener diferentes cargos dentro de la institución.

## 2.7. Segunda Iteración

La segunda iteración viene dada por lo establecido en la historia de usuario 2 (véase tabla 11) en el cual se requiere ver la funcionalidad de parte del usuario con rol de administrador

Dentro del sistema el usuario con rol de administrador será el encargado de administrar todo el sistema, entre sus funciones esta la creación de periodos, materias, cursos, áreas de conocimientos y la administración de usuarios y la gestión de planificaciones.

- Crear usuarios

Para la creación de usuarios habrá que dirigirse al menú de persona y después ir a agregar persona

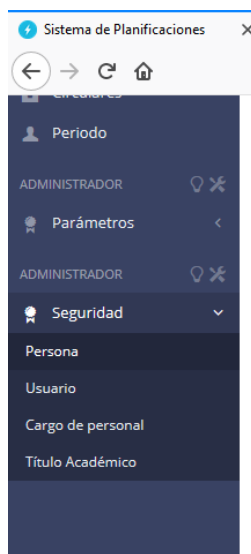


Figura 21. Menú Administrador

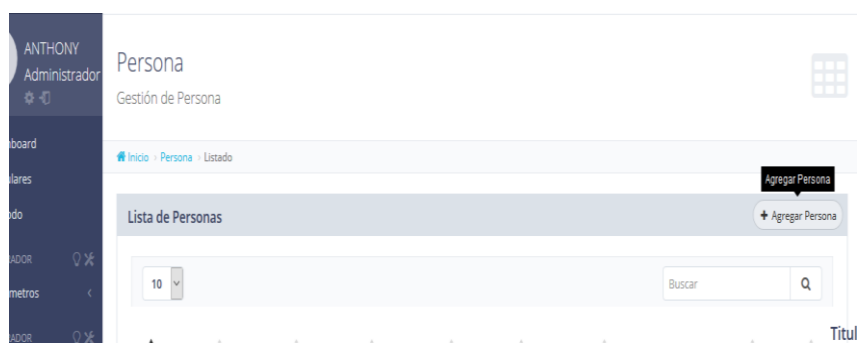


Figura 22. vista personas

Ahora se podrá a llenar la información requerida, al guardar esta persona podrá acceder al sistema como personal

**Formulario Agregar**

**Información Personal**

Cedula \*

Nombres \*

Primer Apellido \*

Segundo Apellido

Teléfono \*

Celular \*

Email \*

Email Opcional

**Información Institucional**

Titulo academico \* SELECCIONAR...

Perfil \* SELECCIONAR...

Figura 23. Formulario persona

- Crear periodo

Para crear un periodo el proceso es similar. Dentro del menú hay que dirigirse a la opción de periodos y agregar un periodo

**UE SPP**

ANTHONY  
Administrador

**Período**  
Gestión de Período

Inicio > Período > Listado

Agregar Período

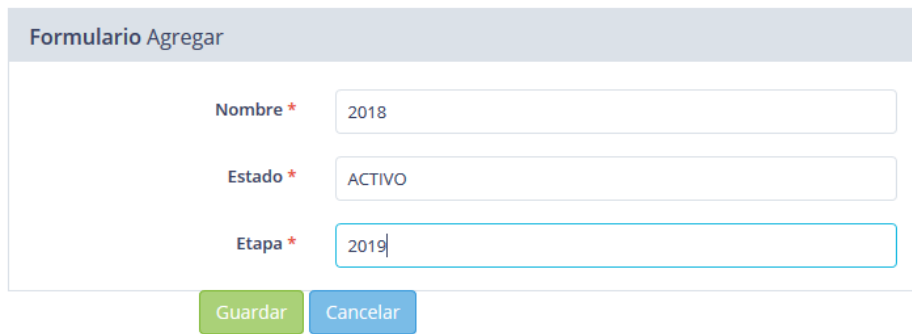
**Lista de Configuración**

10

	Nombre	Estado	Etapas
1	20172018	ACTIVO	

1-1 of 1

Figura 24. Menú Período



**Formulario Agregar**

Nombre \* 2018

Estado \* ACTIVO

Etapa \* 2019

Guardar Cancelar

Figura 25. Formulario periodo

- Crear cursos

En lo que respecta a la creación de cursos se puede establecer tanto un curso y un paralelo.

Para crear cursos hay que dirigirse a la parte de cursos en el menú y agregar un curso



Dashboard  
Circulares  
Periodo  
ADMINISTRADOR  
Parámetros  
Area Conocimiento  
Curso  
Especialidad  
Materia  
Paralelo

Inicio > Curso > Listado

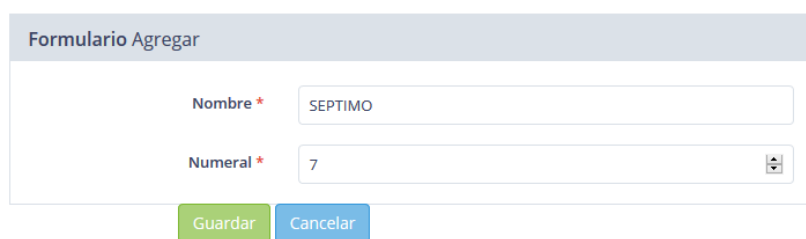
Lista de Curso + Agregar Curso

10 Buscar Q

	Nombre	Numeral	
1	PRIMERO	1	✓ ✕
2	SEGUNDO	2	✓ ✕
3	TERCERO	3	✓ ✕

Figura 26. Menú Curso

Inicio > Curso > Agregar



**Formulario Agregar**

Nombre \* SEPTIMO

Numeral \* 7

Guardar Cancelar

Figura 27. Formulario Curso

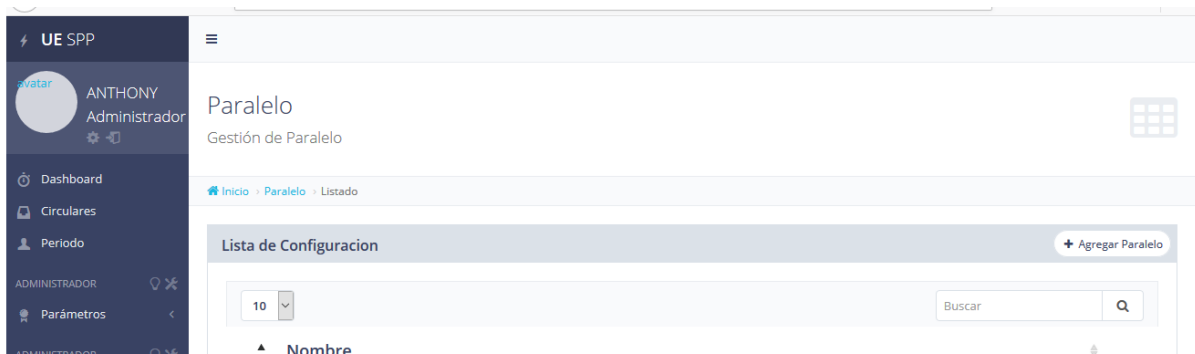


Figura 28. Menú Paralelo

[Inicio](#) > [Paralelo](#) > [Agregar](#)

### Formulario Agregar

**Nombre \***

Guardar
Cancelar

Figura 29. Formulario paralelo

- Crear Área de Conocimiento

Para crear un área de conocimiento hay que dirigirse en el menú a área de conocimiento, agregar área de conocimiento y llenar el formulario.



Figura 30. Menú Área de conocimiento



Figura 31. Formulario Área de Conocimiento

- **Gestión de usuarios**

La gestión de usuario viene dada por la asignación del rol administrador y rector a las personas que se encuentran dentro de la base.

Para eso hay que dirigirse al menú de usuario y agregar un usuario

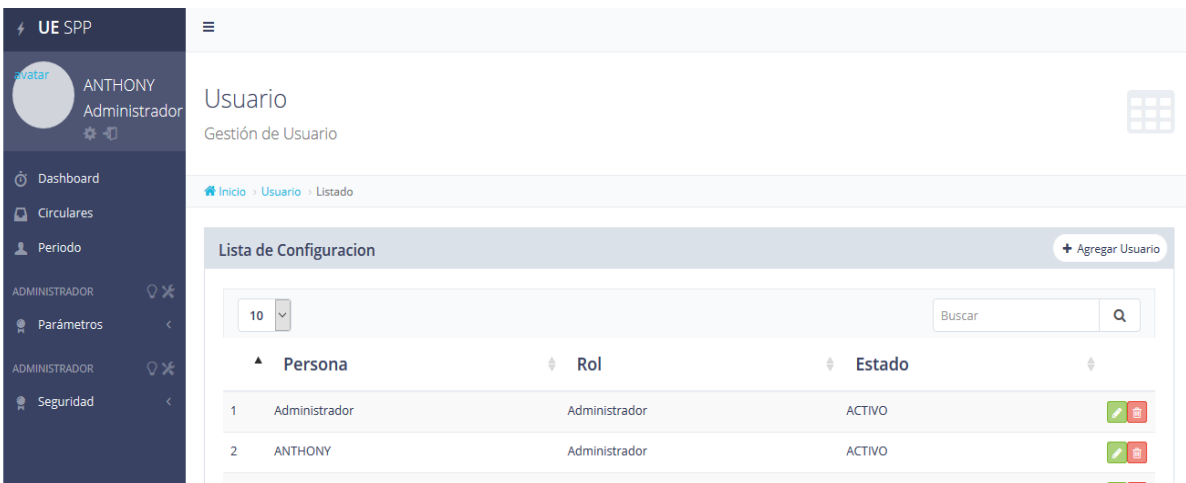


Figura 32. Menú de usuario



Figura 33. Formulario Usuario

Nota: Al momento de crear una persona, este ya podrá ingresar al sistema en calidad de personal, pero para acceder al sistema administrador se requerirá una asignación de rol.

- Gestión de periodo

La gestión del periodo por parte del administrador viene dada por la asignación de cursos y docentes dentro de un periodo, jefes de área a un área de conocimiento y establecer fechas de quimestres y parciales

Para empezar, hay que dirigirse al menú de editar de un periodo

Figura 34. Menú editar periodo

Aquí se puede administrar todo el proceso que conlleva las planificaciones curriculares. En el menú de planificación curricular se encuentra las vistas de docentes que materias dan y en que cursos además de la información acerca de los documentos de planificaciones a los que estarán a cargo.

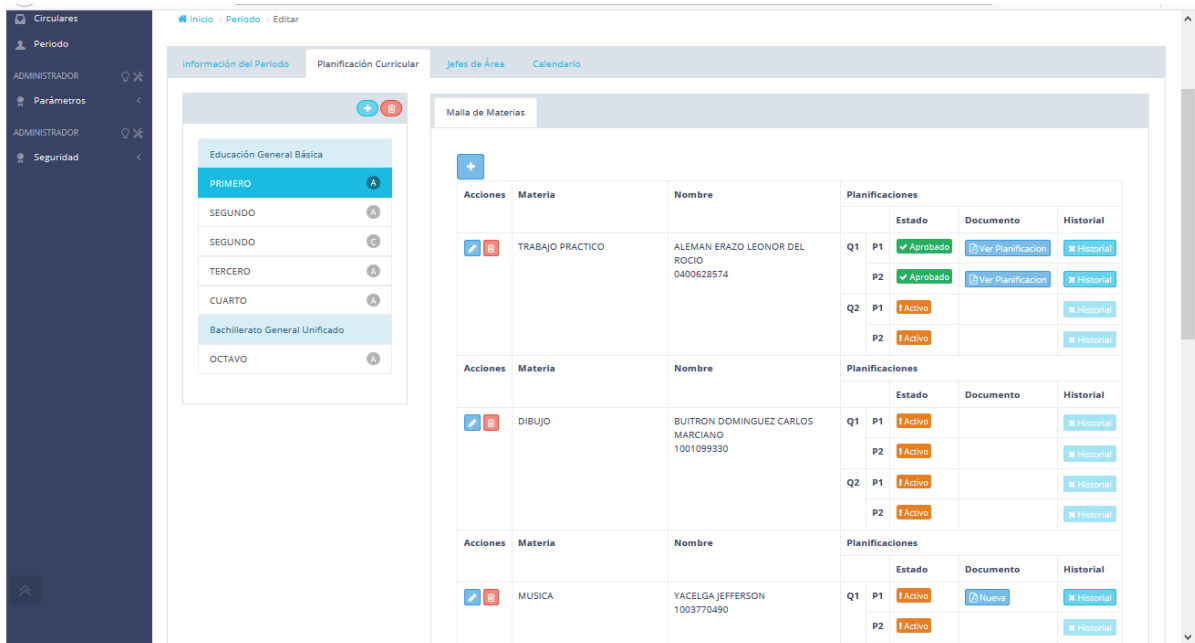


Figura 35. Menú de planificación curricular

Como administrador deberá crear la base para dar inicio al proceso de gestión de planificaciones, con respecto a las planificaciones microcurriculares empiezan al momento de agregar un curso ya que después se podrá asignar un docente con su respectiva materia, para eso hay que ir a la sección donde se muestran los cursos y agregarlos

Figura 36. Formulario para agregar curso

Después de crear un curso se podrán asignar docentes y materia. En la sección de malla de materias dar clic en asignar y llenar el formulario con los datos necesarios.

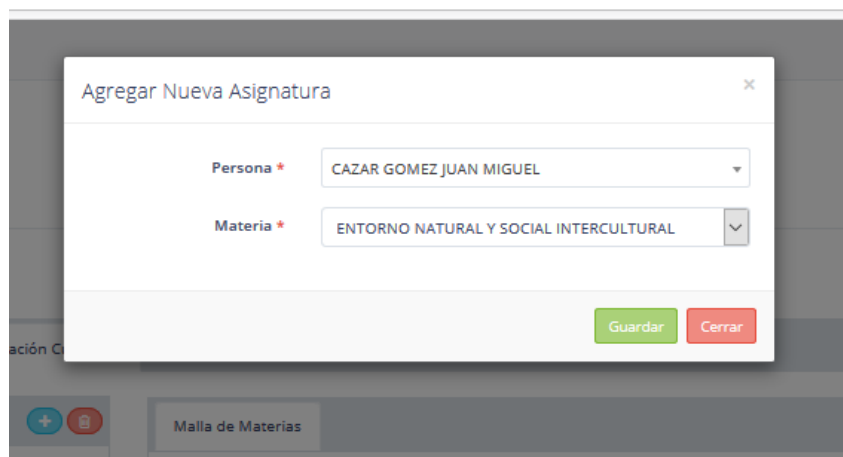


Figura 37. Formulario docente-materia

Nota: Al momento de asignar el docente, la persona asignada se le añadirá la materia de la cual tendrá que subir una planificación, esto se verá reflejado dentro de su cuenta como personal.

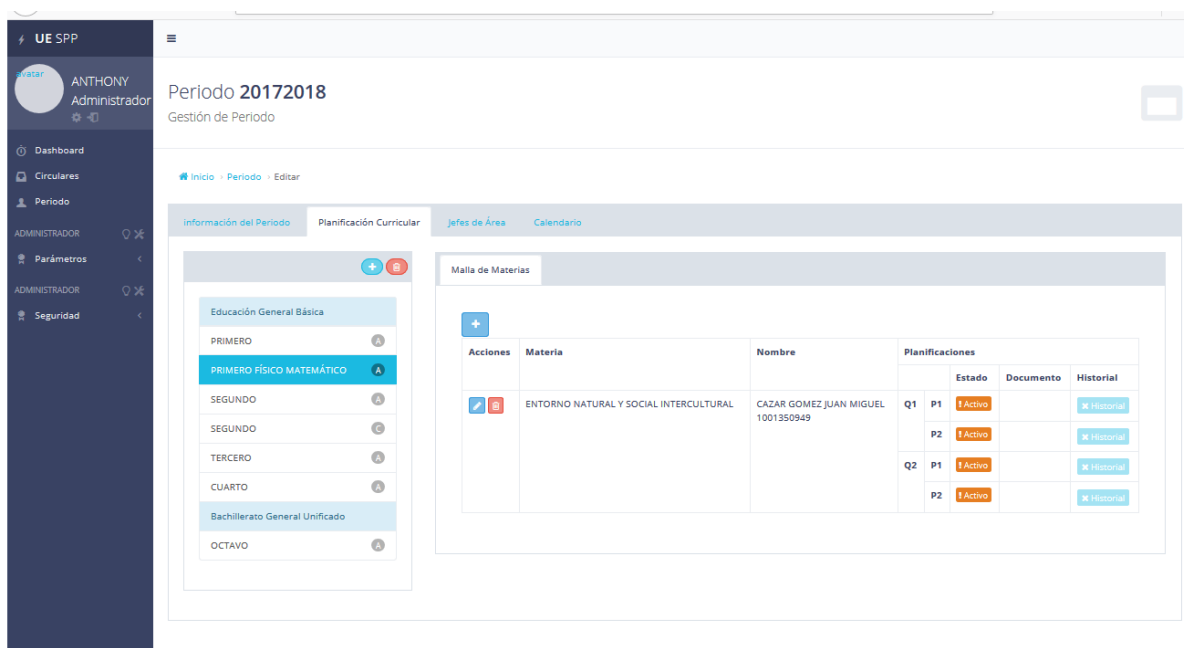


Figura 38. Vista malla de materias después de asignar un docente

Con respecto a los jefes de área ya se mostrarán todas las áreas de conocimientos que maneja la unidad para asignarles un encargado. Basta con asignar una persona.

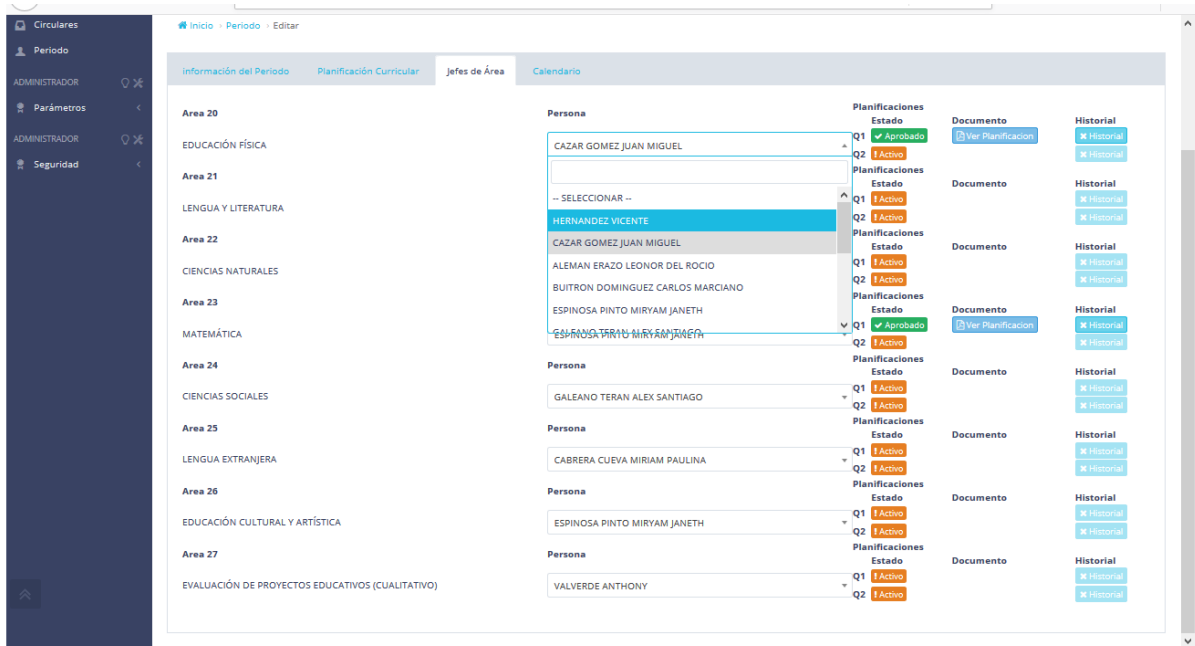


Figura 39. Vista jefe de área

Con respecto a la pestaña Calendario, aquí se podrá establecer la fecha que Para agregar la fecha basta con ir a la barra de fecha y asignar la fecha máxima de entrega

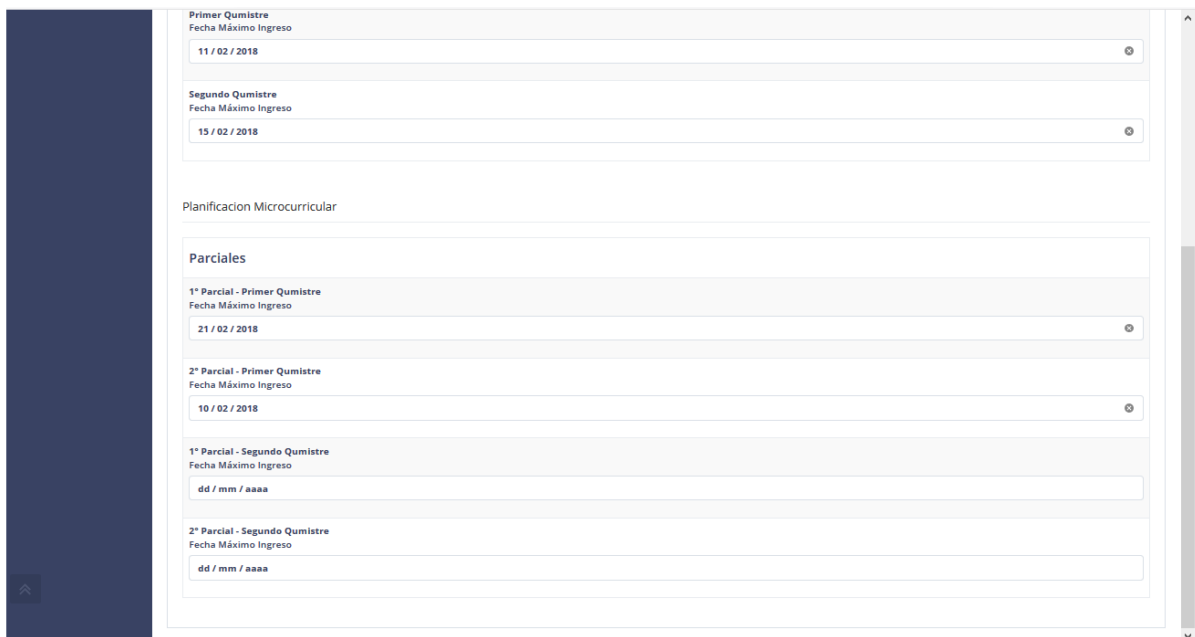


Figura 40. Vista Calendario

## 2.8. Tercera Iteración

La tercera iteración es para cumplir con lo que especifica la historia de usuario 3 (véase la tabla 12), aquí se establecen las funcionalidades a las que podrá acceder el usuario con rol de rector.

Las funciones que tiene el rector son similares a las que posee el administrador, con la diferencia que el podrá visualizar los documento enviados por los docentes aprobarlos o rechazarlos, además de poder enviar circulares a docentes a su correo electrónico y subir la PCI.

- Subir PCI

Dentro de gestión de periodo en la pestaña de información del periodo se encuentra una sección en la que el rector podrá subir la PCI y revisarla cuando él lo requiera

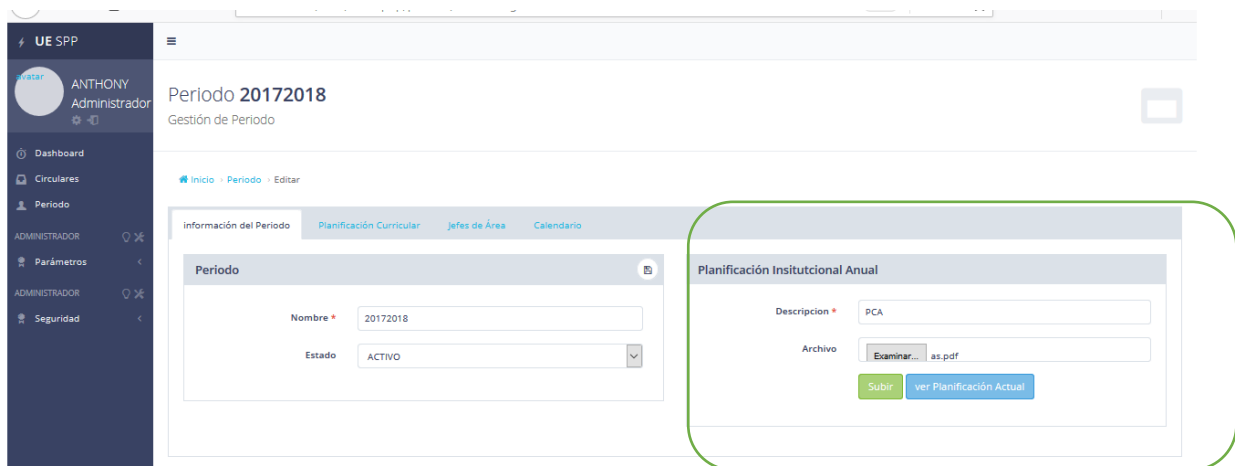


Figura 41. Información del periodo

- Revisión de planificaciones

El proceso de revisión de planificaciones consiste en que, cuando un docente suba una planificación en la parte de gestión de periodo y planificación curricular o jefe de área aparecerá una notificación que dirá nueva, desde esa notificación habrá un enlace directo hacia el archivo para su revisión, después de ser revisado será aprobado o rechazado, en caso de ser aprobado la notificación cambiara y habrá un enlace directo hacia el documento caso contrario desaparecerá hasta el momento que otro documento sea enviado.

				Estado	Documento	Historial
MUSICA	YACELGA JEFFERSON 1003770490	Q1	P1	Activo	Nueva	Historial
			P2	Activo		Historial

Figura 42. Notificación recibo de planificación

Figura 43. Revisión documento

	TRABAJO PRACTICO	ALEMAN ERAZO LEONOR DEL ROCIO 0400628574	Q1	P1	Aprobado	Ver Planificación	Historial
				P2	Aprobado	Ver Planificación	Historial
			Q2	P1	Activo		Historial

Figura 44. Planificación aprobada

Area	Persona	Planificaciones Estado	Documento	Historial
Area 20	CAZAR GOMEZ JUAN MIGUEL	Q1  Aprobado	Ver Planificación	Historial
		Q2  Activo		Historial
Area 21	ALEMAN ERAZO LEONOR DEL ROCIO	Q1  Activo		Historial
		Q2  Activo		Historial
Area 22	BUITRON DOMINGUEZ CARLOS MARCIANO	Q1  Activo		Historial
		Q2  Activo		Historial
Area 23	ESPINOSA PINTO MIRYAM JANETH	Q1  Aprobado	Ver Planificación	Historial
		Q2  Activo		Historial
Area 24	GALEANO TERAN ALEX SANTIAGO	Q1  Activo		Historial
		Q2  Activo		Historial
Area 25	CABRERA CUEVA MIRIAM PAULINA	Q1  Activo		Historial
		Q2  Activo		Historial
Area 26	Persona	Planificaciones Estado	Documento	Historial

Figura 45. Vista planificaciones Jefe de Área

- Envío de circulares

El rector también puede enviar circulares, estos son mensajes que van dirigidos al correo del docente los cuales podrán ser revisados en el portal mismo.

Dentro del menú del sistema hay una opción que se llama circulares, esta sección se asemeja a un correo electrónico, aquí se podrán revisar tanto las circulares recibidas como enviadas y enviar una nueva circular.

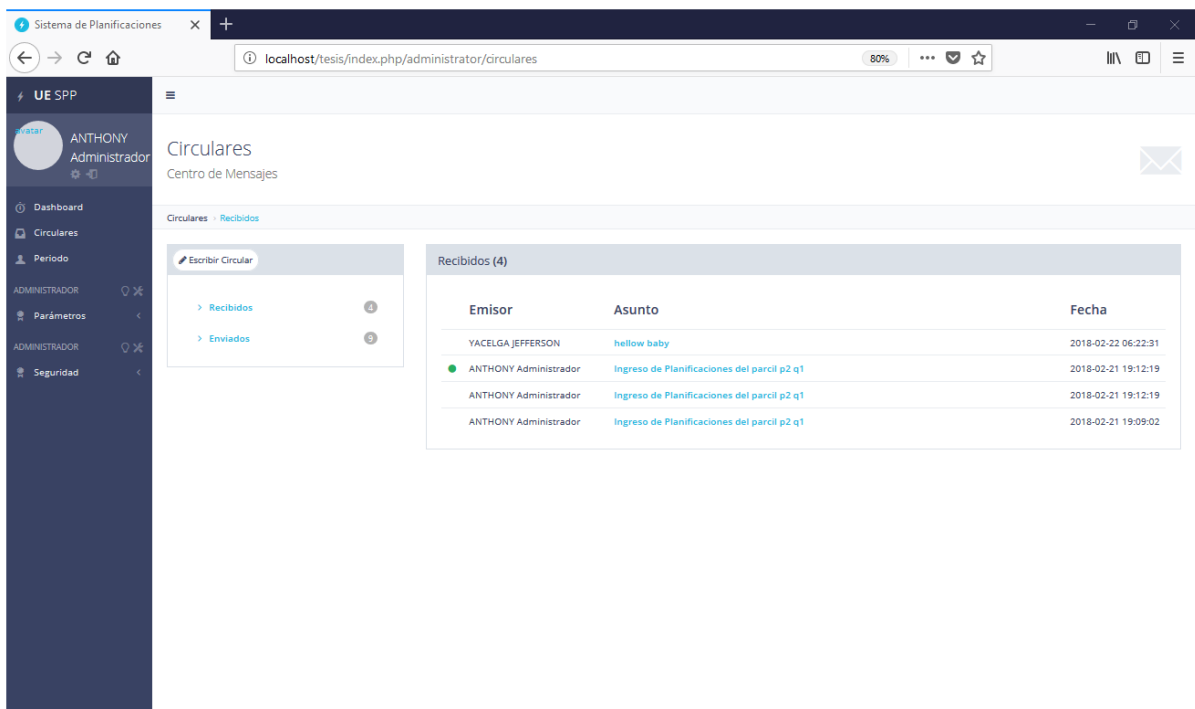


Figura 46. Vista circulares

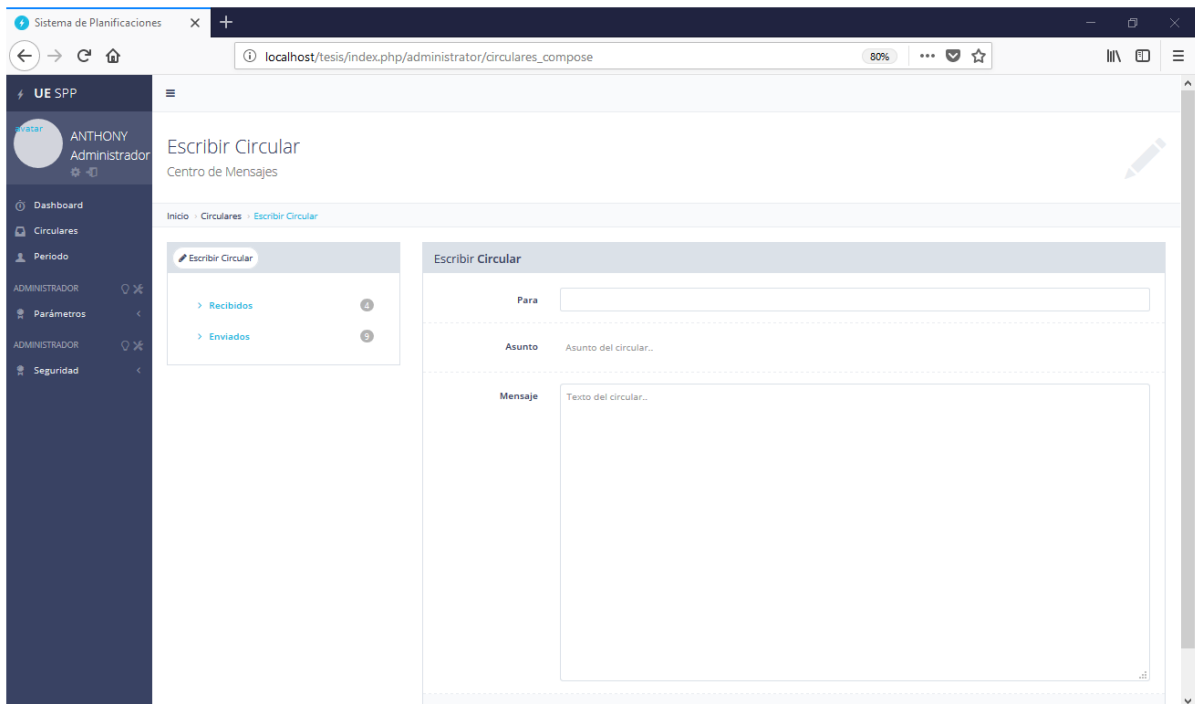


Figura 47. Vista escribir circular

## 2.9. Cuarta Iteración

La cuarta iteración es para cumplir con lo que especifica la historia de usuario 3 y 4 (véase la tabla 12 y 13), las cuales abarcan las funciones de docentes y jefes de área.

Las Funciones que tienen los usuarios designados como docentes o jefes de área son las de enviar las planificaciones correspondientes esperar su aprobación o rechazo y la de recibir los circulares correspondientes

Envío de planificaciones.

Después de ingresar al portal mediante su usuario y contraseña nos encontraremos con la pantalla principal la que nos mostrara las diferentes funciones a las que puede acceder el usuario, como son e ingreso hacia su buzón de circulares, descargar la aplicación móvil del sistema además se mostrara las diferentes materias y áreas de conocimiento a la que el usuario hay sido asignado

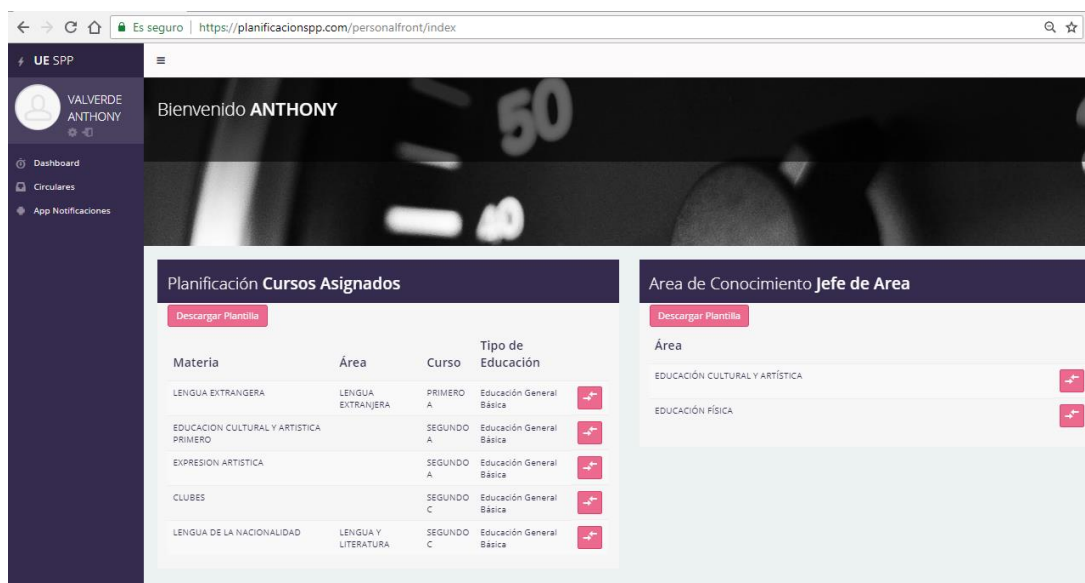


Figura 48. Pantalla principal para usuario docentes.

### Subir Planificación (PCA y planificaciones de aula)

Para subir una planificación el usuario deberá dirigirse al botón que se encuentra al final de cada materia a la que se le haya asignado.

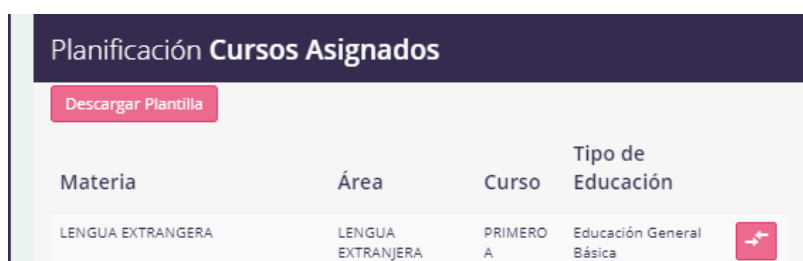


Figura 49. Acercamiento pantalla principal de usuarios docentes

Después nos encontraremos en la pantalla en la que se podrá subir las planificaciones correspondientes.

Para subir una planificación habrá que dirigirse hacia el botón “subir planificación” en la parte del parcial correspondiente



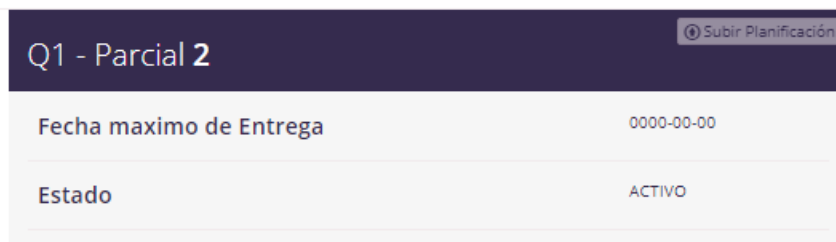


Figura 50. Pantalla principal para usuario docentes

Después subimos el documento

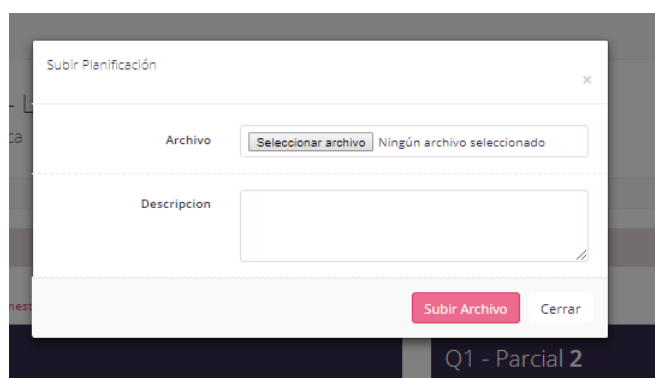


Figura 51. Modal para subir planificación

Se mostrarán las planificaciones que se han subido. Al momento de subir una planificación habrá que esperar la aprobación o rechazo del documento para subir otra planificación si es necesario además se podrán visualizar las observaciones del documento y al momento de darle clic en el icono del documento nos redirigirá hacia un visor pdf para visualizar el documento enviado.

Si el documento fue aprobado terminará el proceso para tal parcial y ya no se podrá subir otro documento.

Nota: El proceso es el mismo para subir las planificaciones por área de conocimiento.

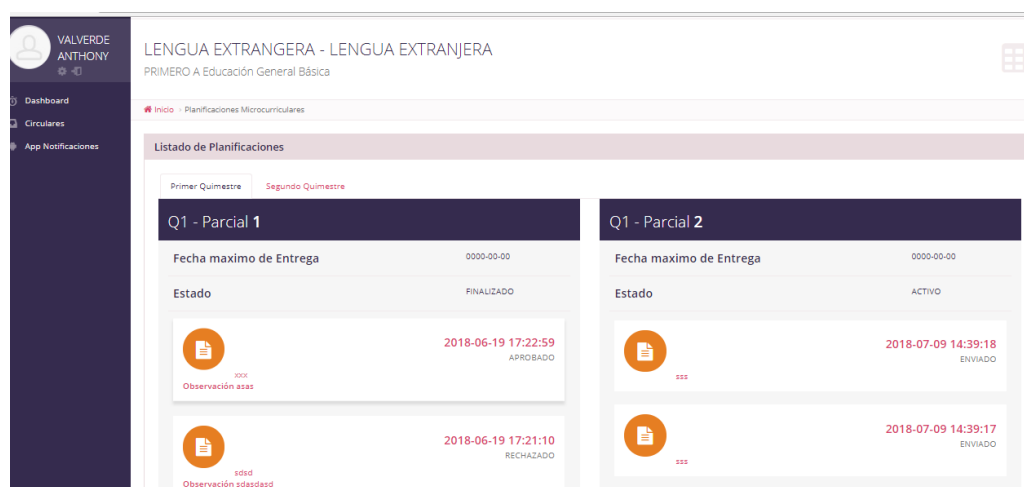


Figura 52. Visualización de documentos enviados

## 2.10. Quinta Iteración

La quinta iteración es para cumplir con lo que especifica la historia de usuario 5 (véase la tabla 14), Esta se refiere al buzón de circulares del sistema.

El buzón de circulares básicamente funciona como un correo de uso interno, mediante este se podrán visualizar y enviar circulares solamente a los usuarios del sistema.

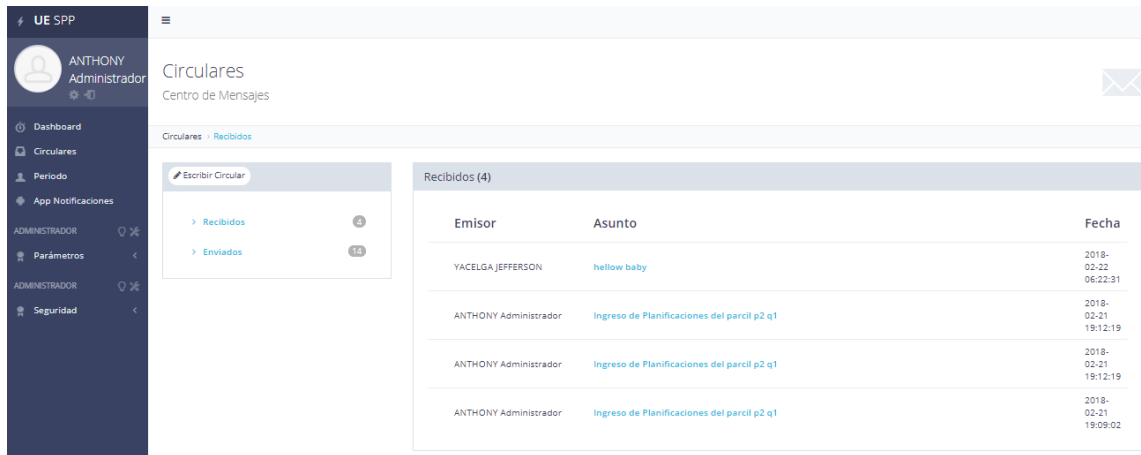


Figura 53. Pantalla de Circulares (Administrador).

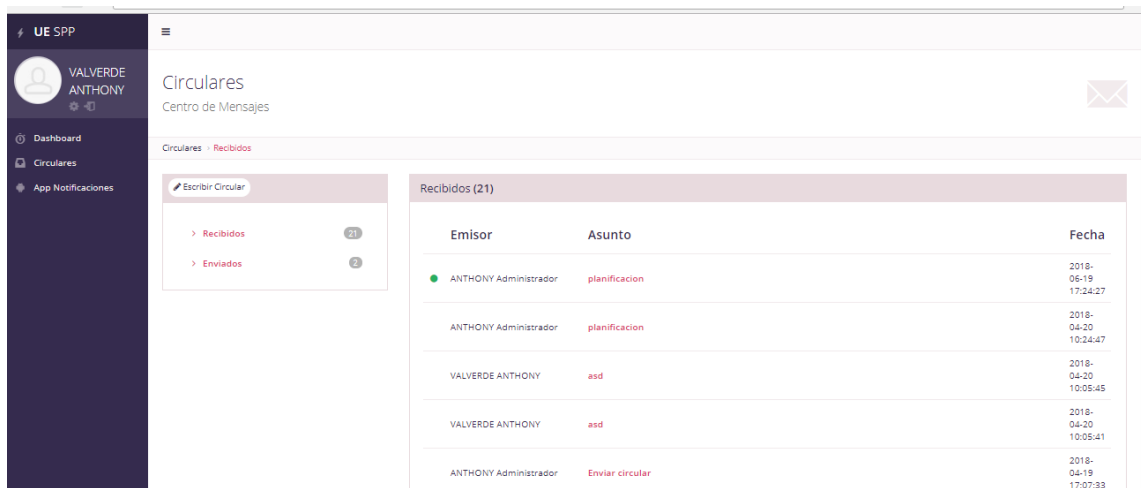


Figura 54. Pantalla de Circulares (Docentes).

La visualización completa de un circular se vera de la siguiente manera.



Figura 55. Visualización completa de un circular (Administrador).

El envío de circulares solo será permitido para los usuarios que estén registrados dentro del sistema.

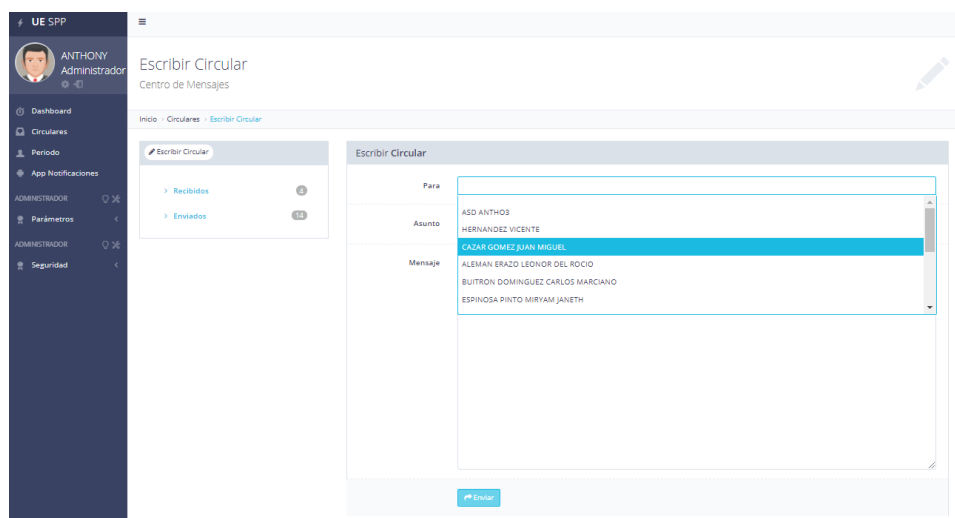


Figura 56. Redacción de un circular

La aplicación para garantizar la revisión de circulares se desarrolló una pequeña aplicación móvil que permitirá a los usuarios revisar las circulares que le han sido enviadas.

Tal aplicación se encuentra disponible dentro del sistema como "App Notificaciones".

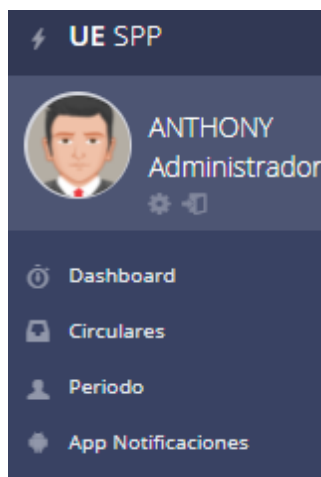


Figura 57. Menú Administrador (App Notificaciones)

El diseño de la aplicación funciona de tal manera que un usuario ingresa con las mismas credenciales del sistema y podrá revisar si tiene nuevas circulares.



Figura 58. Inicio de sesión app

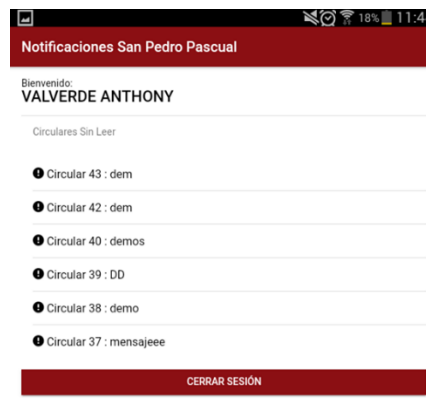


Figura 59. Notificaciones Circular

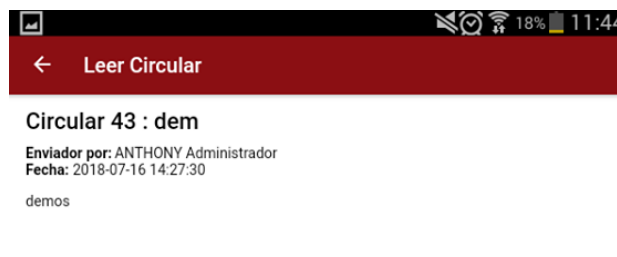


Figura 60. Revisión Circular.

## 2.11. Pruebas de la aplicación

### 2.11.1. Pruebas Unitarias

Las pruebas Unitarias Fueron realizadas y validadas en el documento Anexo “Pruebas de Software”. A continuación, en la tabla 24 se encuentra el detalle de cada prueba realizada.

Tabla 24. Pruebas Unitarias

<b>Nombre de Prueba:</b>	<b>Acceso a base de datos</b>	
<b>Tipo de Prueba:</b>	<input checked="" type="checkbox"/> Unitaria	<input type="checkbox"/> Integración
<b>Objetivo:</b>	Mediante el uso del framework y las respectivas clases de configuración verificar si puede acceder a la base de datos	
<b>Entradas:</b>	Configuración del archivo conf para acceso a la base de datos 'hostname' => 'localhost', 'username' => 'root', 'password' => '', 'database' => 'uessp_planificaciones', 'dbdriver' => 'mysqli',	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Accederá al sistema sin problema	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Crear personas</b>	
<b>Tipo de Prueba:</b>	<input checked="" type="checkbox"/> Unitaria	<input type="checkbox"/> Integración
<b>Objetivo:</b>	Crear una nueva persona	
<b>Entradas:</b>	Información de usuario: Nombre, rol, estado	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Se creará una nueva persona que podrá utilizar el sistema de acuerdo a su rol	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Crear cursos</b>	
<b>Tipo de Prueba:</b>	<input checked="" type="checkbox"/> Unitaria	<input type="checkbox"/> Integración
<b>Objetivo:</b>	Crear cursos vigentes en un periodo	
<b>Entradas:</b>	Información de usuario: Nombre, rol, estado	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Se podrá añadir cursos vigentes dentro de un periodo	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Crear Áreas de Conocimiento</b>	

<b>Tipo de Prueba:</b>	<input checked="" type="checkbox"/> Unitaria	<input type="checkbox"/> Integración
<b>Objetivo:</b>	Crear áreas de conocimiento que maneje el sistema	
<b>Entradas:</b>	Usuarios, áreas de conocimiento	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Se podrá añadir las áreas de conocimiento que maneje el establecimiento	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Subir documento</b>	
<b>Tipo de Prueba:</b>	<input checked="" type="checkbox"/> Unitaria	<input type="checkbox"/> Integración
<b>Objetivo:</b>	Gestionar un periodo	
<b>Entradas:</b>	Usuario, documento	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	El documento se guardará en una carpeta destino	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Envío de Correo</b>	
<b>Tipo de Prueba:</b>	<input checked="" type="checkbox"/> Unitaria	<input type="checkbox"/> Integración
<b>Objetivo:</b>	Notificación mediante circular.	
<b>Entradas:</b>	Usuario Receptor, Mensaje CIMail:: <i>Enviar</i> (\$asunto_email, \$mensaje_email, \$receptor['email_persona'], \$receptor['nombre_nc']);	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	El documento se guardará en una carpeta destino	
<b>Resultado:</b>	True	

## 2.11.2. Pruebas de Integración

Las pruebas de Integración fueron realizadas y validadas en el documento Anexo “Pruebas de Software”. A continuación, en la tabla 25 se encuentra el detalle de cada prueba realizada.

Tabla 25. Pruebas Integración

<b>Nombre de Prueba:</b>	<b>Ingreso al sistema</b>	
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria	<input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Verificar el ingreso del usuario con su respectivo rol.	
<b>Entradas:</b>	Usuario y contraseña	
<b>Salidas:</b>	Afirmativo: True	Negativo: False

<b>Criterio de éxito:</b>	Si el ingreso es correcto el usuario accederá al sistema en donde se mostrará sus respectivas interfaces dependiendo del tipo de usuario.
<b>Resultado:</b>	True
<b>Nombre de Prueba:</b>	<b>Envió de documento</b>
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria <input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Subir planificación al sistema
<b>Entradas:</b>	Usuario y documento
<b>Salidas:</b>	Afirmativo: True Negativo: False
<b>Criterio de éxito:</b>	Si el archivo fue subido con éxito aparecerá un mensaje de confirmación
<b>Resultado:</b>	True
<b>Nombre de Prueba:</b>	<b>Notificaciones de envió dentro del sistema</b>
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria <input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Recibir una notificación al momento que una planificación sea subida al sistema
<b>Entradas:</b>	documento
<b>Salidas:</b>	Afirmativo: True Negativo: False
<b>Criterio de éxito:</b>	Si una planificación fue subida al sistema se mostrará una notificación en el sistema a la persona encargada de revisar el documento
<b>Resultado:</b>	True
<b>Nombre de Prueba:</b>	<b>Notificaciones de respuesta dentro del sistema</b>
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria <input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Recibir una notificación al momento que una planificación sea revisada por la persona encargada
<b>Entradas:</b>	documento
<b>Salidas:</b>	Afirmativo: True Negativo: False
<b>Criterio de éxito:</b>	La persona responsable de la planificación encontrara una notificación al momento de que su documento sea revisado, con sus respectivas observaciones
<b>Resultado:</b>	True
<b>Nombre de Prueba:</b>	<b>Envió de circulares</b>
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria <input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Enviar un correo a los usuarios, en caso de ser necesario
<b>Entradas:</b>	nombre usuario, asunto, texto

<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Un mensaje llegara a los correos, teléfonos y aplicación de los usuarios destinatarios	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Gestión de un periodo</b>	
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria	<input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Gestionar un periodo	
<b>Entradas:</b>	Usuario, rol	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	La persona encargada podrá gestionar y revisar todo el proceso de envío de planificaciones en un periodo activo	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Asignar Docente - Curso- Materia</b>	
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria	<input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Asignar una materia a un docente en su respectivo curso	
<b>Entradas:</b>	Usuario, rol, materia, curso, periodo	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Se mostrará un docente dentro del curso con su respectiva materia	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Asignar Jefe de Área- Área de conocimiento</b>	
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria	<input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Asignar una materia a un docente en su respectivo curso	
<b>Entradas:</b>	Usuario, rol, materia, curso, periodo	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Se mostrará un docente dentro del curso con su respectiva materia	
<b>Resultado:</b>	True	
<b>Nombre de Prueba:</b>	<b>Reportes de planificaciones</b>	
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria	<input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Generar reportes tanto de planificaciones por área como por materia.	
<b>Entradas:</b>	Usuario, rol, materia, curso, periodo, área de conocimiento	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Se mostrará un reporte en pdf del estado del proceso de envío de planificaciones curriculares.	
<b>Resultado:</b>	True	



<b>Nombre de Prueba:</b>	<b>Reportes de Docentes</b>	
<b>Tipo de Prueba:</b>	<input type="checkbox"/> Unitaria	<input checked="" type="checkbox"/> Integración
<b>Objetivo:</b>	Generar reportes de docentes registrados en el sistema	
<b>Entradas:</b>	Usuario, rol.	
<b>Salidas:</b>	Afirmativo: True	Negativo: False
<b>Criterio de éxito:</b>	Se mostrará un reporte en pdf de todos los docentes que estén registrado en el sistema.	
<b>Resultado:</b>	True	

# CAPITULO III

## IMPLEMENTACIÓN

### 3.1. Instalación.

La instalación del sistema requiere del contrato de un dominio a un servidor de paga en el cual se subirá toda la aplicación y la base de datos, esto no requiere de mayor conocimiento por parte del administrador.

### 3.2 Requisitos.

Los requisitos para usar el sistema web son los siguientes.

- Acceso a Internet
- Navegador web (Chrome, explorer, firefox)
- Dispositivo móvil inteligente.
- Servidor de paga

Los requisitos para la aplicación móvil son los siguientes.

- Dispositivo móvil Android
- Acceso a internet

#### 3.2.1 Requisitos de servidor

Al momento de contratar un dominio hay que revisar que el servidor cuento con los siguientes requisitos para el perfecto funcionamiento del sistema.

Tabla 26. Requisitos Servidor

<b>Lenguaje de programación</b>	PHP 7.0
<b>SGBD</b>	MySQL
<b>Espacio de disco duro</b>	10gb o superior
<b>Memoria RAM</b>	1GB o superior
<b>Servicio de correo</b>	Activado
<b>Disponibilidad</b>	Alta

### 3.3 Alojamiento

El sistema estará alojado dentro de un servidor de paga el cual contendrá un dominio en donde se subirá la aplicación con todos los paquetes que CodeIgniter tiene, además de subir

también la base de datos y su información. Al momento de importar la base de datos al servidor es posible que requiera hacer ciertas modificaciones dentro del backup de la base de datos a subir, al igual que en la aplicación se tendrá configurar el archivo de configuración de la base de datos por la nueva información.

### 3.4 Despliegue

Una vez se haya subido la aplicación al servidor el despliegue consta en probar el dominio, cargando el enlace en los diferentes navegadores y probar el funcionamiento del sistema. Para mostrar el funcionamiento del sistema se utiliza lo que son las pruebas de aceptación, estas serán realizadas por el cliente para constatar el funcionamiento del sistema.

### 3.5 Matriz de valoración de entrega de documento.

Con el fin de constatar que el sistema ha mejorado el proceso de gestión de planificaciones en la Institución con respecto al tiempo, entrega y revisión del documento se ha realizado una matriz para valorar la tarea de entrega y revisión de documento ya que esto es una de las tareas más importantes que realiza el sistema. Los parámetros evaluados se los califico como Mejora (el proceso mejoro), Normal (El proceso se mantiene igual), Malo (El proceso se volvió más complicado). En la tabla 27 se observa a detalle la matriz.

<b>Parámetro de Valoración</b>	<b>Resultado</b>
Entrega de documento por parte del personal docente	Mejora
Revisión del documento por parte de las personas autorizadas	Mejora
Mejora en el tiempo de entrega del documento	Mejora
Seguridad de envío y recepción de documento	Mejora
Acceso a documentos enviados de manera individual	Mejora

Tabla 27 Matriz de valoración

### 3.6. Pruebas de Aceptación

Una vez el sistema se encuentra ya hospedado dentro de un dominio, resta hacer las pruebas de aceptación para comprobar que el sistema funciona a la perfección. Tales pruebas se verán reflejadas como resultados de cada historia de usuario y de funcionalidades

planteadas dentro del proyecto. La tabla 28 en la que se muestra los resultados fue sacada del documento Anexo “Pruebas de Software”.

Tabla 28.Pruebas Aceptación

Historia de usuario	Tarea	Resultado
<b>H1. Acceso al Portal</b>	El acceso al sistema solo es permitido a personas que estén registradas dentro de la base de datos y se filtrara funcionalidades de acuerdo al rol asignado	Éxito
<b>H2. Gestión del portal (Usuario: rector, administrador)</b>	Administrar Usuarios	Éxito
	Administrar Personas	Éxito
	Administrar Materias	Éxito
	Administrar Áreas de conocimiento	Éxito
	Administrar Periodos	Éxito
	Administrar Planificaciones	Éxito
<b>H3. Gestión de Circulares y Planificaciones. (Usuario: rector, administrador)</b>	Revisión de Planificaciones	Éxito
	Aprobación/Rechazo de planificaciones	Éxito
	Subir Planificación (PCI)	Éxito
	Envió de Circulares	Éxito
<b>H4. Publicar los PCA y Enviar Circulares.</b>	Subir PCA	Éxito
	Recibir Notificación de Aprobación o Rechazo	Éxito
	Enviar Circular.	Éxito
<b>H5. Publicar Planificación de Aula y Enviar Circulares</b>	Subir Planificación de Aula	Éxito
	Recibir Notificación de Aprobación o Rechazo	Éxito
	Enviar Circular.	Éxito
<b>H6. Buzón de Circulares</b>	Enviar Circular	Éxito
	Revisión de Circular	Éxito
<b>Herramientas</b>	Integración de herramientas Libres	Éxito
<b>Seguridad</b>	Base de datos	Éxito
	Encriptación de información (contraseñas, documentos.)	Éxito
	Creación dinámica de contraseñas.	Éxito

	Manejo del Sistema de Tipo Administrador.	Éxito
<b>Valoración de entrega de documentos</b>	Mejora tiempo de entrega del documento	Éxito
	Seguridad de envío y recepción de documento	Éxito
	Reporte a detalle de entregas	Éxito
	Acceso a documentos enviados de manera individual	Éxito

## CONCLUSIONES

- La implementación de la aplicación mediante el uso del framework CodeIgniter dio como resultado un sistema con una estructura organizada, un desarrollo rápido por la forma en la que maneja el uso del lenguaje de programación PHP, mejoró el uso del lenguaje de programación mediante el uso de nuevas funciones y un manejo de comunicación entre librerías que usa el framework.
- La implementación del sistema optimizó de manera considerable el proceso de gestión de planificaciones curriculares dentro de la institución, habiendo mejoras en aspectos como eficacia, mejora en la seguridad, envío de documentos, revisión y aprobación de los diferentes documentos que son utilizados en el proceso de planificaciones, con lo que resulta un desarrollo del proceso más organizado, seguro y con un estricto seguimiento por parte de las personas encargadas de revisar los documentos lo que obliga a los docentes a cumplir con este proceso en un tiempo limitado, cumpliendo a cabalidad con el proceso de manera más rápida.
- El uso de la metodología XP ayudó a llevar a cabo esta investigación de manera estructurada y organizada por lo que se logró documentar e implementar un sistema eficiente eficaz y de calidad para el usuario en un tiempo establecido.
- La integración de herramientas libres para este proyecto dio como resultado un producto de calidad, el cual puede ser visualizado tanto en ordenadores como en dispositivos móviles utilizando buenas prácticas de desarrollo, una programación estructurada y un producto visualmente amigable, fácil de usar para el cliente, además de no escatimar en el gasto de licencias para el uso del sistema lo que fue una motivante dentro de la institución para optar por el sistema, por lo que la institución solo tendrá que costar gastos de alojamiento.

## RECOMENDACIONES

- Al momento de escoger un framework para realizar un proyecto hay que revisar cuales son las herramientas con las que trabaja y de qué manera lo hace, para familiarizarse con el modo en que trabaja el framework y así se vuelva una ayuda al momento de desarrollar mas no una carga para el desarrollo del proyecto,
- Fomentar el uso de herramientas libres mediante cursos o conferencias en los cuales se indiquen las ventajas que implican en manejar herramientas libres al momento de desarrollar utilizando como ejemplo sistemas sencillos de realizar en ese instante, para que así los estudiantes se sientan más motivados a utilizar cierta herramienta.
- Tener conocimientos acerca de la forma en la que trabajan varias metodologías conocer como mínimo el ciclo de desarrollo de la metodología para así tener una idea de cómo se puede poner en marcha un proyecto y cuál sería la mejor metodología por usar.
- Conocer a cabalidad el proceso que se va a automatizar, conocer cuáles son las personas involucradas dentro del proceso y las tareas que realizan, ya que son tareas que se verán reflejas en funcionalidades del sistema, es bueno diseñar un diagrama del proceso.

## BIBLIOGRAFIA

- Alvarez, M. A. (2012). Manual de CodeIgniter.
- Ana Flores, J. M. (2014). Buscando la excelencia educativa: Gestión de procesos académicos y administrativos en Instituciones Públicas de Educación mediante BPM.
- Bahit, E. (2012). Scrum & extremeProgramming. Buenos Aires, Argentina: safeCreative.
- Delfa, L. N., Galdámez, N., Thomas, P. J., & Pesado, P. M. (2013). Un Análisis Experimental de Tipo de Aplicaciones para dispositivos móviles. Argentina.
- Erich, G., Richard, H., Johnson, R., & Vlissides, J. (2012). Patrones de diseño. Addison Wesley.
- Fontela, C. (2013). UML. alfaomega.
- Garita-Araya, R. A. (2013). Tecnología Móvil: desarrollo de sistemas y aplicaciones. E-Ciencias de la Información.
- IBM. (2012). El desarrollo de aplicaciones móviles nativas, Web o híbridas. Estados Unidos.
- INEC. (2017). INEC. Obtenido de <http://www.ecuadorencifras.gob.ec>
- IT, s. (2018). db-engines.com. Obtenido de <https://db-engines.com/en>
- MINEDUC. (2016). INSTRUCTIVO PARA PLANIFICACIONES CURRICULARES PARA EL SISTEMA NACIONAL DE EDUCACIÓN. Quito.
- Navarrete, E. N. (2017). Desarrollo de un marco de trabajo (framework) para el desarrollo de aplicaciones web en la Universidad Nacional de Costa Rica.
- Pantaleo. (2015). Ingeniería de Software. Alfaomega.
- Phan, H. (November 2016 ). Ionic 2 Cookbook Second Edition.
- RENAUX, J. (2017). Ionic 2 vs. Ionic 1: Performance Gains, New Tools, and a Big Step Forward.



- Revilla, E. (2017). Construye Apps móviles multiplataforma con ionic 2 desde cero.
- Talón, E. M. (2016). Apache. aulamentor.
- Torres, M. (2014). Desarrollo de aplicaciones web con PHP y MySQL. Lima: Macro.
- Velo, F. (2012). CodeIgniter Guía del Usuario en Español.
- Wood, D. (2015). Diseño de Interfaces. UBEdicio.
- Yenisleidy, R., & Yanette, G. (2012). Patrón Modelo-Vista-Controlador.
- Yolanda, B. L. (recuperado noviembre 2017). Metodología Ágil de Desarrollo de Software – XP.

## **Anexos**

Anexo 1: Documento de Pruebas de Software (en CD).