

PROTOTIPO FUNCIONAL DE APLICATIVO MÓVIL PARA LA UNIVERSIDAD
CATÓLICA DE PEREIRA PARA LA ADMINISTRACIÓN DE INFORMACIÓN
PERSONAL DEL ESTUDIANTE

RICHARD ALEXANDER DUQUE DURANGO
HERMAN LEONARDO ALARCÓN SIERRA

UNIVERSIDAD CATÓLICA DE PEREIRA
FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍA
INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES
PEREIRA
2016

PROTOTIPO FUNCIONAL DE APLICATIVO MÓVIL PARA LA UNIVERSIDAD
CATÓLICA DE PEREIRA PARA LA ADMINISTRACIÓN DE INFORMACIÓN
PERSONAL DEL ESTUDIANTE

RICHARD ALEXANDER DUQUE DURANGO
HERMAN LEONARDO ALARCÓN SIERRA

Presentado a:
DANIEL BLANDON

UNIVERSIDAD CATÓLICA DE PEREIRA
FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍA
INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES
PEREIRA
2016

Nota de Aceptación

Presidente del Jurado

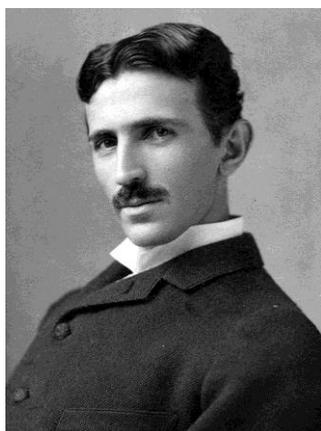
Jurado

Jurado

Este proyecto y su contenido son de autoría de Richard Alexander Duque Durango y Herman Leonardo Alarcón Sierra.

Se autoriza que el proyecto sea utilizado como base de consulta para la Universidad Católica de Pereira y sus estudiantes y que pueda ser modificado si solo si se incluyen mejoras o servicios nuevos y teniendo en cuenta las siguientes condiciones:

- No se eliminará ningún servicio construido.
- No se utilizará para lucrarse económicamente.
- El aplicativo es para la comunidad académica de la Universidad Católica de Pereira y así debe de continuar.
- No se extraerá código o información para la realización de otros proyectos que estén fuera de la universidad.
- Si es modificado, se deben dar créditos a las personas que iniciaron con el proyecto.
- Si es modificado, se seguirá con los estándares documentados para código, diseño, versión entre otros.
- Si es modificado, es para la mejora del aplicativo.
- Si es modificado, se deberá de actualizar manual de usuario y documento de ingeniería de software.



“Si tuviera la suerte de alcanzar alguno de mis ideales sería en nombre de toda la
humanidad.”
NIKOLA TESLA

AGRADECIMIENTOS

A NUESTROS FAMILIARES

A nuestros familiares que siempre nos apoyaron en todo momento y nos enseñaron los buenos valores, a nuestros amigos que siempre nos dieron una mano.

A nuestros docentes que nos educaron y siempre nos orientaron de la mejor forma, a la universidad católica de Pereira que nos formó para ser gente, gente de bien y profesionalmente capaz.

Lista de Contenido

| | | |
|--------|--|----|
| 1.0. | CAPÍTULO 1 | 18 |
| 1.1. | PLANTEAMIENTO DEL PROBLEMA..... | 18 |
| 1.2. | OBJETIVOS | 19 |
| 1.2.1. | Objetivo General | 19 |
| 1.2.2. | Objetivos Específicos | 19 |
| 1.3. | JUSTIFICACIÓN | 20 |
| 1.4. | ALCANCE..... | 20 |
| 1.5. | VIABILIDAD DEL PROYECTO | 20 |
| 1.6. | VIABILIDAD FINANCIERA | 22 |
| 1.7. | CONSECUENCIAS DE LA INVESTIGACIÓN..... | 22 |
| 1.6.1. | VENTAJAS:..... | 22 |
| 1.6.2. | DESVENTAJAS:..... | 23 |
| 1.7. | APORTE PRÁCTICO | 23 |
| 1.8. | APORTE TEORÍCO..... | 23 |
| 1.9. | FACTIBILIDAD | 24 |
| 1.9.1. | Factibilidad Técnica | 24 |
| 1.9.2. | Factibilidad Operativa | 24 |
| 1.9.3. | Factibilidad Económica. | 24 |
| 2.0. | CAPITULO 2 | 25 |
| 2.1. | ANTECEDENTES | 25 |
| 2.1.1. | UN PS | 25 |
| 2.1.2. | Unilibre..... | 25 |
| 2.1.3. | Fundación CIDCA | 25 |
| 2.1.4. | UTP Móvil | 26 |
| 2.1.5. | UCP Mobile –Pretrópolis Brasil..... | 26 |
| 2.2. | MARCO CONTEXTUAL..... | 27 |
| 2.2.1. | A nivel local | 27 |
| 2.2.2. | Reseña Histórica | 27 |
| 2.2.3. | Población Objeto de Estudio | 30 |
| 2.2.4. | Misión..... | 30 |

| | |
|--|----|
| 2.2.5. Visión..... | 31 |
| 2.3. MARCO CONCEPTUAL | 32 |
| 2.3.1. Ingeniería de Software | 32 |
| 2.3.2. Programación Web | 32 |
| 2.3.3. Paradigma de la Programación Orientada a Objetos..... | 32 |
| 2.3.4. FrameWork Bootstrap | 34 |
| 2.3.5. Base De Datos..... | 34 |
| 2.3.6. FrameWork .Net | 35 |
| 2.3.7. Programación Por Capas..... | 35 |
| 2.3.8. Metodología Ágil Mobile-D..... | 36 |
| 2.3.9. AndroidStudio | 37 |
| 2.3.10. Microsoft VisualStudio..... | 38 |
| 2.3.11. SQLITE | 38 |
| 2.3.12. SDK..... | 38 |
| 2.3.13. Microsoft SQL SERVER | 39 |
| 2.3.14. Oracle | 39 |
| 2.3.15. Normalización | 39 |
| 2.3.16. XML | 40 |
| 2.3.17. JAVA | 40 |
| 2.3.18. C# | 40 |
| 2.3.19. Webservice..... | 41 |
| 2.3.20. Seguridad Webservice | 42 |
| 2.3.21. Protocolo HTTPS | 42 |
| 2.3.22. Servidor Web (Internet Information Service) | 43 |
| 2.3.23. GIT..... | 43 |
| 2.3.24. Team Foundation Server | 43 |
| 2.3.25. Window Communication Foundation | 44 |
| 2.3.26. Diagramas UML | 44 |
| 2.3.27. HTML | 45 |
| 2.3.28. ASP.NET | 45 |
| 2.3.29. Web Service REST | 46 |
| 2.3.30. Oracle 11G..... | 46 |

| | |
|--|-----|
| 2.3.31. Librería Gson | 46 |
| 2.3.32. Librería DBFlow..... | 47 |
| 2.3.33. Librería Retrofit 2 | 47 |
| 2.3.34. Librería Volley..... | 47 |
| 2.4. ESPECIFICACIÓN DE LA SOLUCIÓN | 48 |
| 2.4.1. Requerimientos | 48 |
| 2.4.2. UML | 51 |
| 2.4.3. Nomenclatura..... | 76 |
| 2.4.4. Esquemas de datos | 77 |
| 2.4.5. Interfaz de usuario | 80 |
| 2.4.6. Plantilla de pruebas | 83 |
| 3.0. CAPITULO 3 | 84 |
| 3.1. METODOLOGÍA..... | 84 |
| 3.1.1. Se aplicó ingeniería del software al proyecto. | 86 |
| 3.1.2. Realizar un estudio sobre arquitectura de software e implementar dicha arquitectura para tener una mejor estructuración del proyecto. | 86 |
| 3.1.3. Diseñar prototipo de software para un fácil manejo basándonos en encuestas. | 87 |
| 3.1.4. Codificar de acuerdo a lo encontrado en el objetivo 1. | 91 |
| 3.1.5. Realizar pruebas al software para garantizar su funcionamiento. | 91 |
| 3.1.6. Implantar versión beta en Google Play Store | 91 |
| 3.2. PLANIFICACIÓN DEL PROYECTO | 92 |
| 3.2.1. GANTT | 92 |
| 3.2.2. PERT..... | 95 |
| 3.3. CRONOGRAMA | 96 |
| 3.4. PRESUPUESTO..... | 99 |
| RECOMENDACIONES | 101 |
| CONCLUSIONES..... | 102 |
| APÉNDICES: | 103 |
| APÉNDICE A: | 103 |
| APÉNDICE B: | 103 |
| BIBLIOGRAFÍA | 104 |

Lista de Tablas

| | |
|--|-----|
| Tabla 1 – Actores | 52 |
| Tabla 2 - Registrar Dispositivo (CU_REQ_01) | 53 |
| Tabla 3 - Borrar y Sincronizar (CU_REQ_02) | 54 |
| Tabla 4 - Cambiar Idioma (CU_REQ_03) | 55 |
| Tabla 5 - Información Personal (CU_REQ_04)..... | 56 |
| Tabla 6 - PIN o Contraseña (CU_REQ_05)..... | 58 |
| Tabla 7 - Vincular cuenta estudiante (CU_REQ_06) | 59 |
| Tabla 8-Agregar o cambiar contraseña (CU_REQ_07) | 60 |
| Tabla 9-Contraseña invalidad (CU_REQ_08) | 60 |
| Tabla 10-Versión Android --- IMEI (CU_REQ_09)..... | 61 |
| Tabla 11-Vinculación, Sincronización y GPS --- IMEI (CU_REQ_10)..... | 62 |
| Tabla 12-Vinculación, Sincronización y GPS --- IMEI (CU_REQ_10)..... | 63 |
| Tabla 13 - ConsultarHorarios_Calificaciones_Libros (CU_REQ_11) | 64 |
| Tabla 14 - Noticias (CU_REQ_12) | 65 |
| Tabla 15 - verHimno_Factultad_Programas (CU_REQ_13)..... | 65 |
| Tabla 16 - verHimno_Factultad_Programas (CU_REQ_13)..... | 66 |
| Tabla 17 - Ayuda (CU_REQ_14)..... | 66 |
| Tabla 18 - Información UCP y ayuda al usuario..... | 69 |
| Tabla 19 - Sincronización GPS, IMEI y consultas | 71 |
| Tabla 20 - Información personal opciones vinculación de estudiante | 73 |
| Tabla 21 - Sincronización y verificación de usuario y contraseña..... | 75 |
| Tabla 22 - Componentes de diseño | 76 |
| Tabla 23 – Variables | 76 |
| Tabla 24 - Base de datos..... | 77 |
| Tabla 25 - Interfaz de usuario | 82 |
| Tabla 26 - Ficha técnica encuesta diseño fácil manejo. | 87 |
| Tabla 27 - Cronograma | 98 |
| Tabla 28 - Presupuesto..... | 100 |

Lista de Imágenes

| | |
|--|----|
| Imagen 1 - Carta de aprobación | 21 |
| Imagen 2 - Infografía Reseña Historia P1 | 27 |
| Imagen 3 - Infografía Reseña Historia P2 | 28 |
| Imagen 4 - Infografía Reseña Historia P3..... | 29 |
| Imagen 5 - Programación por capas..... | 35 |
| Imagen 6 - Metodología D-Mobile | 36 |
| Imagen 7 – Funcionamiento WebService | 41 |
| Imagen 8 - Características herramienta UML..... | 45 |
| Imagen 9 - EDT Proyecto UCP-PIM | 48 |
| Imagen 10 - Registrar Dispositivo (CU_REQ_01) | 53 |
| Imagen 11 - Borrar y Sincronizar (CU_REQ_02)..... | 55 |
| Imagen 12 - Cambiar Idioma (CU_REQ_03)..... | 56 |
| Imagen 13 - Información Personal (CU_REQ_04) | 57 |
| Imagen 14 - PIN o Contraseña (CU_REQ_05) | 58 |
| Imagen 15-Vincular cuenta estudiante (CU_REQ_06) | 59 |
| Imagen 16-Agregar o cambiar contraseña (CU_REQ_07)..... | 60 |
| Imagen 17-Contraseña invalidad (CU_REQ_08)..... | 61 |
| Imagen 18-Versión Android --- IMEI (CU_REQ_09) | 62 |
| Imagen 19 – Consultar Horarios Calificaciones Libros (CU_REQ_11) | 64 |
| Imagen 20 - Noticias (CU_REQ_12) | 65 |
| Imagen 21 - Ayuda (CU_REQ_14)..... | 67 |
| Imagen 22 - Información UCP y ayuda al usuario | 68 |
| Imagen 23 - Sincronización GPS, IMEI y consultas | 70 |
| Imagen 24 - Información personal opciones vinculación de estudiante..... | 72 |
| Imagen 25 - Sincronización y verificación de usuario y contraseña | 74 |
| Imagen 26 - Esquema Conceptual | 77 |
| Imagen 27 - Entidad Relación Cliente | 78 |
| Imagen 28 - Entidad Relación Servidor (Prototipo) | 79 |
| Imagen 29 - Modelo Relacional Servidor (Prototipo)..... | 80 |
| Imagen 30 - Interfaz - Registrar | 81 |
| Imagen 31 - Interfaz - Ingresar..... | 81 |
| Imagen 32 - Interfaz - Horarios | 81 |
| Imagen 33 - Interfaz - Notas | 81 |
| Imagen 34 - Interfaz - Noticias | 82 |
| Imagen 35 - Interfaz - SIABUC | 82 |

| | |
|--|----|
| Imagen 36 - Interfaz - Mi Universidad..... | 82 |
| Imagen 37 - Interfaz - Acerca de | 82 |
| Imagen 38 - Mobile-D | 84 |
| Imagen 39 - Pregunta 1 Encuesta..... | 87 |
| Imagen 40 - Pregunta 2 Encuesta..... | 88 |
| Imagen 41 - Pregunta 3 Encuesta..... | 88 |
| Imagen 42 - Pregunta 4 Encuesta..... | 89 |
| Imagen 43 - Pregunta 5 Encuesta..... | 89 |
| Imagen 44 - Pregunta 6 Encuesta..... | 90 |
| Imagen 45 - Pregunta 7 Encuesta..... | 90 |
| Imagen 46 – Cronograma GANTT 01..... | 92 |
| Imagen 47 – Cronograma GANTT 02..... | 92 |
| Imagen 48 – Cronograma GANTT 03..... | 93 |
| Imagen 49 – Cronograma GANTT 04..... | 94 |
| Imagen 50 - Cronograma PERT | 95 |

Glosario

UCP: Universidad Católica de Pereira

PIM: Personal Information Manager (Administración de la Información Personal)

POO: Programación Orientada a Objetos

CIAF: Corporación Instituto de Administración y Finanzas (CIAF):

UNITECNICA: Universidad Técnica

CIDCA: Centro de Investigación Docencia y Consultoría Administrativa

UTP: Universidad Tecnológica de Pereira

UniLibre: Universidad Libre

UN PS: Universidad Nacional Sistema de Posicionamiento

FN: Forma Normal

Resumen

Este proyecto comienza en la Universidad Católica De Pereira observando al estudiante diariamente detectando así sus dificultades.

El proyecto UCP-PERSONAL INFORMATION MANAGER (UCP-PIM) parte de un proceso de conceptualización que inicia con una dificultad detectada en la interacción con compañeros de clase y docentes, la cual consiste en tener un canal de información más ágil y pertinente que permita consultar en tiempo real algunos aspectos académicos de interés general para todos. Como resultado de esto se decide desarrollar un App que pueda suplir las necesidades del usuario de que la información sea móvil.

Se consultó en internet y en la tienda de Google información sobre proyectos parecidos de diferentes universidades; pero solo en universidades de la región.

La Universidad Tecnológica de Pereira cuenta con un App en la tienda de Google; Universidades como la Universidad Libre de Pereira, Fundación Universitaria del Área Andina, Universidad Cooperativa de Colombia e institutos tales como IDONTEC, CIAF, UNITECNICA no cuentan con aplicativos móviles.

Se encontró que el Instituto de educación Superior CIDCA cuenta con un aplicativo que al parecer no es funcional ya que cuenta con mala calificación y comentarios negativos.

El objetivo principal de este proyecto es la creación e implementación de una aplicación móvil universitaria, que sirva como herramienta de ayuda al estudiante para facilitar algunos procedimientos que realizan diariamente en la Universidad.

Se espera como parte del resultado, que el usuario final ahorre tiempo y además de ello, que se le facilite y sea más amigable su estadía en la Universidad Católica de Pereira.

Palabras Claves: Universidad, Instituto, Estudiante, Aplicativo, Herramienta, Necesidades, Dificultad, Información.

ABSTRACT

This project begins at the Catholic University of Pereira, watching student the daily to thereby detecting their difficulties.

The UCP-PERSONAL INFORMATION MANAGER (UCP-PIM) project starts from a conceptualization process that begins with a difficulty detected in the interaction with classmates and teachers, which consists of having a more agile and relevant information channel that allows consulting In real time some academic aspects of general interest for all. As a result of this it is decided to develop an App that can meet the needs of the user that the information is mobile.

We search the internet and the Google store for information on similar projects from different universities; But only at universities in the region.

The Technological University of Pereira has an App in the Google store; Universities like the Free University of Pereira, Andean Area University Foundation, Cooperative University of Colombia and institutes such as IDONTEC, CIAF and UNITECNICA do not have mobile applications.

It was found that the Institute of Higher Education CIDCA has an application that apparently is not functional as it has poor rating and negative comments.

The main objective of this project is the creation and implementation of a mobile university application, which serves as a tool to help the student to facilitate some procedures that they perform daily in the University.

It is expected as part of the result, that the end user saves time and in addition, it will be facilitated and more friendly his stay in the Catholic University of Pereira.

Keywords: University, Institute, Student, Application, Tool, Needs, difficulty, information.

INTRODUCCIÓN

Desde el primer sitio web en 1991 (elmundo.es, 2015) las Tecnologías de la Información se han convertido en el pan de cada día de toda gran empresa; En la actualidad no existe empresa grande y ordenada que no tenga su propio software para gestionar y facilitar la información a las personas dentro y fuera de la compañía.

En los inicios del siglo XXI las Tecnologías de Información han tenido cambios muy notorios e impactantes para el usuario final, uno de ellos es el de los dispositivos móviles inteligentes, según (Alonso, 2015) cada mes se crean aproximadamente 263.000 App en la tienda de Google y desaparecen cerca de 43.000, por ende en las siguientes páginas se va a dar a conocer una herramienta amigable y con funcionalidades que brindarán al estudiante de la Universidad Católica de Pereira, información personal y de la Universidad, Todo esto sólo en su bolsillo.

La Universidad Católica de Pereira es un establecimiento educativo de educación superior con más de 43 años educando y brindando conocimiento a la comunidad. La UCP siempre está en busca del mejoramiento de la calidad para los estudiantes, actualmente es la Universidad con más programas acreditados de la región. Cuenta con servicios para que los estudiantes estén al tanto de lo que pasa en la Universidad Católica de Pereira, con sus notas y horarios y la búsqueda de libros pero algunos de estos servicios que presta están divididos en diferentes sistemas y en ocasiones es demorado acceder a estos.

El presente proyecto busca ofrecerle a la Universidad un producto software donde se puedan centralizar varios servicios desde un aplicativo móvil, agilizando la búsqueda de información personal del estudiante.

1.0. CAPÍTULO 1

1.1. PLANTEAMIENTO DEL PROBLEMA

El presente proyecto UCP-PIM (Universidad Católica de Pereira-Personal Information Manager), se propone por la dificultad que tiene el estudiante para acceder a su información personal y realizar procedimientos necesarios diariamente en el establecimiento educativo.

Todo comienza desde el primer día en el que se ingresó a estudiar a la Universidad Católica de Pereira, se notó desde el principio lo monótono que es sacar el carnet de la billetera diariamente para registrar un ingreso, Desde ese día se empezaron a notar detalles como tener que sacar el cuaderno para ver el horario de clase, escribir o llamar a un compañero para pedirle el salón de clase, preguntar al docente la sobre las calificaciones de X asignaturas, no saber lo que pasa en la UCP y hasta tener que esperar a que desocupen un equipo para buscar un libro en la biblioteca.

Se observó que el estudiante de la UCP por lo general desea tener acceso rápido, fácil y sin necesidad de estar siempre conectado a internet para ver su información, incluso sin encender su ordenador, sobre las facultades, programas, biblioteca, calificaciones, horarios, en las siguientes páginas detallamos el proceso de desarrollo de dicho software.

1.2. OBJETIVOS

1.2.1. Objetivo General

Desarrollar prototipo funcional de aplicativo móvil para la Universidad Católica de Pereira sujeto a buenas prácticas de programación para que el estudiante pueda tener acceso a notas, horarios, noticias, SIABUC e información sobre la Universidad.

1.2.2. Objetivos Específicos

- 1) Aplicar ingeniería del software a este proyecto
- 2) Realizar un estudio sobre arquitectura de software e implementar dicha arquitectura.
- 3) Diseñar prototipo de software para un fácil manejo basándonos en encuestas realizadas utilizando el formulario de Google y un Muestreo Aleatorio Probabilístico.
- 4) Codificar de acuerdo a lo encontrado en el objetivo específico 1.
- 5) Realizar pruebas al software para garantizar su buen funcionamiento
- 6) Implantar versión beta del software en GooglePlay Store.

1.3. JUSTIFICACIÓN

Este proyecto busca ofrecer información del estudiante y de la universidad católica de Pereira en tiempo real con mayor agilidad, eficiencia, eficacia, confiabilidad, facilidad, seguridad y sobre todo algo único entre el resto de universidades de la región que facilite al estudiante las tareas de consulta diarias más relevantes. Por el lado de la universidad se evitarán peticiones al servidor de aplicaciones y por ende al servidor de base de datos, mejorando el GoodWill de la universidad.

Todo ello se logrará por medio de UCP-PIM, un aplicativo para que los estudiantes utilicen este recurso y así facilitar diariamente su estadía en la Universidad Católica de Pereira.

1.4. ALCANCE

Este proyecto fue desarrollado para ofrecer cuatro servicios.

Los cuatro servicios que contendrá el prototipo funcional serán:

- Servicio SIABUC
- Servicio Calificaciones
- Servicio Horarios
- Servicio Noticias

Los 3 servicios restantes que no fueron incluidos en este alcance son:

- Servicio Apuntes post it
- Servicio Ingreso UCP
- Servicio Comunidad

1.5. VIABILIDAD DEL PROYECTO

Teniendo en cuenta todo lo necesario para el desarrollo de este aplicativo móvil UCP-PIM, se evidencia que se tiene a favor todos los recursos necesarios para el proyecto.

Las herramientas para el desarrollo de este software tales como IDEs, compiladores, documentación, motores de base de datos, apoyo de docentes y estudiantes con conocimiento básico para iniciar son más que suficientes.

En la universidad Católica de Pereira se cuentan con muy buenos recursos tecnológicos y calidad en docentes para el apoyo en este proceso.

Se cuenta con la aprobación de la universidad y la necesidad de incluirla en los sistemas de información de la institución.

| | | | | |
|---|---|---------|---------------|-------------|
|  | Fecha de elaboración del formato: 22 de mayo de 2008 | ID: 804 | Página 1 de 1 | SGC-AR-1-02 |
|---|---|---------|---------------|-------------|

VA-300-22

Pereira, Agosto 26 de 2016

Apreciados estudiantes
RICHARD ALEXANDER DUQUE DURANGO
HERMAN LEONARDO ALARCON SIERRA
Universidad Católica de Pereira

Cordial saludo.

Para la Universidad Católica de Pereira resulta muy loable que ustedes quieran realizar su proyecto UCP-PIM con el objetivo de desarrollar un prototipo funcional de aplicativo móvil para nuestra institución. Comprenderán que nuestros sistemas de notas son de uso exclusivo de nuestro personal autorizado dada la gran responsabilidad y confidencialidad de la información que se maneja.

La Universidad quiere colaborar con su proyecto y les ofrece poner a su disposición un sistema de información alterno con el cual ustedes puedan realizar las pruebas y ejercicios para avanzar en su trabajo, para tal efecto se pueden poner en contacto con el Ing. Daniel Felipe Blandón. Espero que el ofrecimiento sea de su total satisfacción y cumpla con sus expectativas.

Atentamente,

WILLMAR DE JESUS ACEVEDO GOMEZ
Vicerrector Académico

Universidad Católica de Pereira - Avenida de la América, Carrera 21 No. 49-93 A.A.2453 411221-403-791-1
PBF 512 4000 FAX 512 76 15 E-MAIL uca@uca.edu.co Web-site: www.uca.edu.co - Pereira - Santander - Colombia - S.A.

| | |
|---|--|
| Estado: Controlado Sistema Control de Calidad | Asesor: Director Administrativo y Financiero |
|---|--|

Imagen 1 - Carta de aprobación

1.6. VIABILIDAD FINANCIERA

El proyecto es viable financieramente porque es viable socialmente ya que con la aparición de los Smartphone todo se está unificando en un solo lugar y el usuario cada vez usa menos una computadora.

También es viable financieramente porque la Universidad Católica de Pereira implementando este prototipo funcional mejorará el GoodWill y será pionera a nivel municipal como la primer Universidad en tener un Aplicativo Móvil propio para facilitar los procesos que realizan los estudiantes.

1.7. CONSECUENCIAS DE LA INVESTIGACIÓN

Con el prototipo funcional UCP-PIM, la Universidad tendrá un entendimiento completo y requerimientos claros sobre el funcionamiento de un aplicativo móvil para la Universidad Católica de Pereira.

Por medio de este prototipo funcional, la Universidad Católica de Pereira tendrá la posibilidad de continuar con el desarrollo de UCP-PIM y realizar la implementación de este App para facilitar al estudiante los procesos que realizan diariamente en la Universidad tales como Consulta de Notas, Horarios, Noticias y catálogos de libros(SIABUC), ya que tendrán la información el bolsillo y al momento de consultar información de notas y horarios no se verán obligados a encender el computador, todo será más rápido y con la posibilidad de no estar conectado a internet.

1.6.1. VENTAJAS:

- El estudiante tendrá más facilidad de acceso a su información personal universitaria.
- El estudiante ahorrará tiempo en varios procesos que se realiza diariamente en la Universidad como la consulta de notas, horario, búsquedas de información en SIABUC y el portal de noticias.
- Disminución del pool de conexiones al servidor de notas y horarios
- La rapidez será una de las principales ventajas de UCP-PIM porque tendrá la posibilidad de funcionar en modo offline.

1.6.2. DESVENTAJAS:

- Solo funcionará en plataformas Android
- Funcionará desde la versión 4 de Android en adelante.

1.7. APORTE PRÁCTICO

El aporte logrado con esta aplicación es poner en contacto a los estudiantes con la plataforma web que tiene la universidad, con ello la reducción de tiempo a la hora de consultas y estando al tanto de las noticias y actividades de la universidad, además tendrá un referente tecnológico: Una App móvil que en el momento no cuenta con ella.

Los mayores beneficiarios son los estudiantes ya que podrán hacer consultas más rápidamente con el aplicativo móvil y estarán en constante conocimiento sobre las diversas noticias y actividades que realiza la universidad.

El proyecto busca que los estudiantes disminuyan el tiempo de respuesta cuando deciden consultar sus notas, horarios, noticias y consultas bibliográficas en la biblioteca. Con el App lo podrán realizar de forma rápida, con lo que evitan estar desplazándose hasta la biblioteca para saber qué libro está disponible o en qué ubicación se encuentra solo con consultar en la App, así podrán saber la sección para una mayor facilidad.

1.8. APORTE TEORÍCO

Este proyecto siguió un desarrollo en el cual se aplicó ingeniería de software y una metodología de desarrollo enfocada a móviles llamada Mobile-D.

Una de las formas para escribir un código mantenible fue inicialmente definir una nomenclatura y una arquitectura para facilitar y agilizar el desarrollo tanto en el AndroidStudio para el aplicativo móvil como para el Webservice en VisualStudio. Para el desarrollo del webservice se utiliza una arquitectura en 5 capas de las cuales las que están en constante modificación es solo la capa de Entidades y la Capa de Regla de Negocios.

Para el desarrollo móvil se aplica una metodología recomendada para el desarrollo móvil llamada Mobile-D, La cual agiliza el desarrollo móvil en proyectos no tan grandes.

1.9. FACTIBILIDAD

Se ha tenido en cuenta la factibilidad del proyecto en los tres aspectos siguientes:

1.9.1. Factibilidad Técnica

Se cuenta con la tecnología necesaria para el desarrollo del prototipo funcional ya que se tienen al alcance equipos para el desarrollo y configuración de los mismos para las pruebas del prototipo, También se cuenta con programas adecuados para la codificación, Además se cuenta con conocimientos de programación.

1.9.2. Factibilidad Operativa

Basados en encuestas a nivel mundial, se decide que el proyecto es factible a nivel operativo ya que este suplirá una necesidad del estudiante de la UCP, y teniendo en cuenta que un gran porcentaje de ellos trabajan con sistema operativo Android en sus teléfonos se concluye que a nivel operativo el proyecto es factible.

Según un estudio desarrollado por Strategy Analytics, un 87,5% de los nuevos teléfonos vendidos a lo largo del tercer trimestre de 2016 funciona con Android. Con las cifras delante, eso significa que de los 375 millones de smartphones, 328 millones son de Android. Apple solo tiene un 12,1% de la cuota total del mercado, unos 45,5 millones de iPhones. (EL MUNDO, 2016)

Por otro lado la interacción que se va a brindar en la aplicación es factible para el usuario, ya que su fácil uso le permitirá interactuar mejor con la interfaz, logrando con esto una reducción de tiempo en las distintas búsquedas que deben realizar.

Actualmente la Universidad cuenta con una Red WLAN adecuada para el uso del aplicativo por parte del estudiante, Teniendo en cuenta que el aplicativo no siempre necesitará estar conectado a una red con internet.

1.9.3. Factibilidad Económica

El equipo del proyecto tuvo en cuenta las herramientas de uso libre que se utilizaran en el desarrollo, horas de ingeniero de desarrollo, de pruebas, por tal motivo es factible a nivel económico, además de ello se ha tenido en cuenta otros gastos que se pueden suplir por el equipo y los cuales son descritos en el capítulo 3 punto 3.4. Presupuesto.

2.0. CAPITULO 2

2.1. ANTECEDENTES

En la actualidad según las consultas realizadas no existe un Aplicativo móvil totalmente funcional para una de las universidades o institutos del sector. A continuación, proyectos de otras universidades del país o del exterior, similares o idénticos al planteado en el objetivo general de este documento:

2.1.1. UN PS

Autor y año del proyecto: OC Developers - 2014

Título del proyecto: UN PS

Resumen: Aplicación para estudiantes, docentes, administrativos y visitantes de la Universidad Nacional de Colombia Sede Bogotá. Funciona para trazar rutas peatonales entre edificios y locaciones importantes de la universidad sin necesidad de una conexión de internet, además brinda un listado completo de los edificios de la sede Bogotá con su historia y una sencilla interfaz para crear tu horario. (OC Developers, 2014)

2.1.2. Unilibre

Autor y año del proyecto: Universidad Seccional Barranquilla - 2015

Título del proyecto: Unilibre

Resumen: Aplicativo móvil con las siguientes funcionalidades: Notificaciones con las últimas novedades, información institucional, descripción de la ubicación por sedes de servicios, programas académicos, admisiones, acceso a SIUL, educación virtual, buzón de sugerencias entre otros. (Universidad Libre Seccional Barranquilla, 2015)

2.1.3. Fundación CIDCA

Autor y año del proyecto: Jaahser, jaahser@hotmail.com - 2015

Título del proyecto: Fundación CIDCA

Resumen: Consulta Horario y Notas Fundación CIDCA (Ser, 2015)

2.1.4. UTP Móvil

Autor y año del proyecto: Gestión de TI y SI - 2015

Título del proyecto: UTP Móvil

Resumen: Mantente conectado con Universidad Tecnológica de Pereira donde quiera que estés. La aplicación UTP móvil, contiene funcionalidades públicas y privadas. Tiene funciones tales como: Visualizar las Noticias del campus informa, Encuentra los teléfonos y extensiones en el Directorio telefónico institucional, Ubicarse geográficamente con el Mapa del campus Universitario, Enviar sugerencias de la App a nuestro buzón, Acceso al portal estudiantil, Revise sus Notas parciales, Visualice su Historial académico, Conozca los Horarios profesores y aulas asignados. (Gestión TI y SI, 2015)

2.1.5. UCP Mobile –Pretrópolis Brasil

Autor y año del proyecto: Bruno Priori - 2016

Título del proyecto: UCP Mobile (Universidad Católica de Petrópolis-Brasil)

Resumen: Tener en su mano toda la información en su Universidad. La UCP móvil lleva a los estudiantes toda la información académica de la Universidad Católica de Petrópolis, como los horarios de clase, eventos de calendario, el acceso al Estudiante virtual, entre otras funciones. (Priori, 2016)

2.2. MARCO CONTEXTUAL

2.2.1. A nivel local

Los dispositivos móviles han cambiado la manera de comunicarnos con las personas ya que con estos realizamos muchas actividades tales como llamadas telefónicas, mensajes de texto chats, video llamadas, acceso a información en internet entre otros, cosas que antes se hacían con diferentes dispositivos.

Las Tecnologías de la Información han ayudado a satisfacer diferentes necesidades y mantener informado a las personas, por dicho motivo la población de la investigación es la universidad Católica de Pereira.

2.2.2. Reseña Histórica

Las App Store han cambiado nuestro concepto de software y aplicaciones. En la actualidad es extraño ver que se entregue el software con CD's, además porque algunos ordenadores ya no se venden con unidad de CD-ROM. En lugar de todo eso, ya se consiguen los programas a través de las tiendas online. La era del Smartphone ha propiciado la proliferación de las tiendas de aplicaciones. A continuación, se muestra una infografía donde se describe una breve historia sobre las tiendas online.

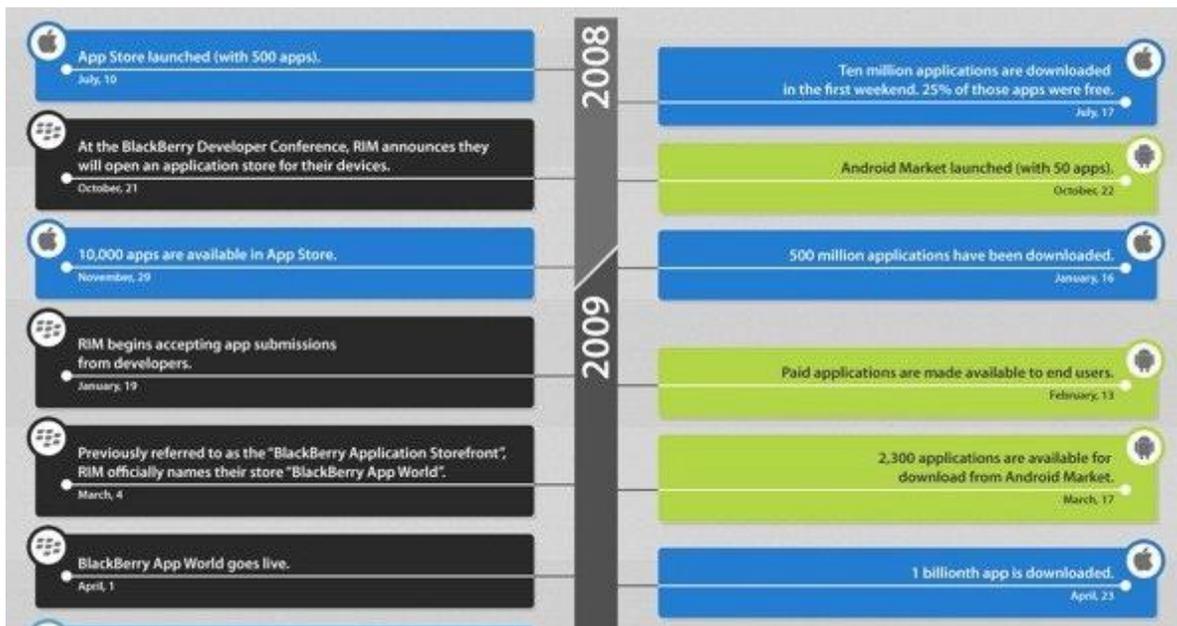


Imagen 2 - Infografía Reseña Historia P1



Imagen 3 - Infografía Reseña Historia P2

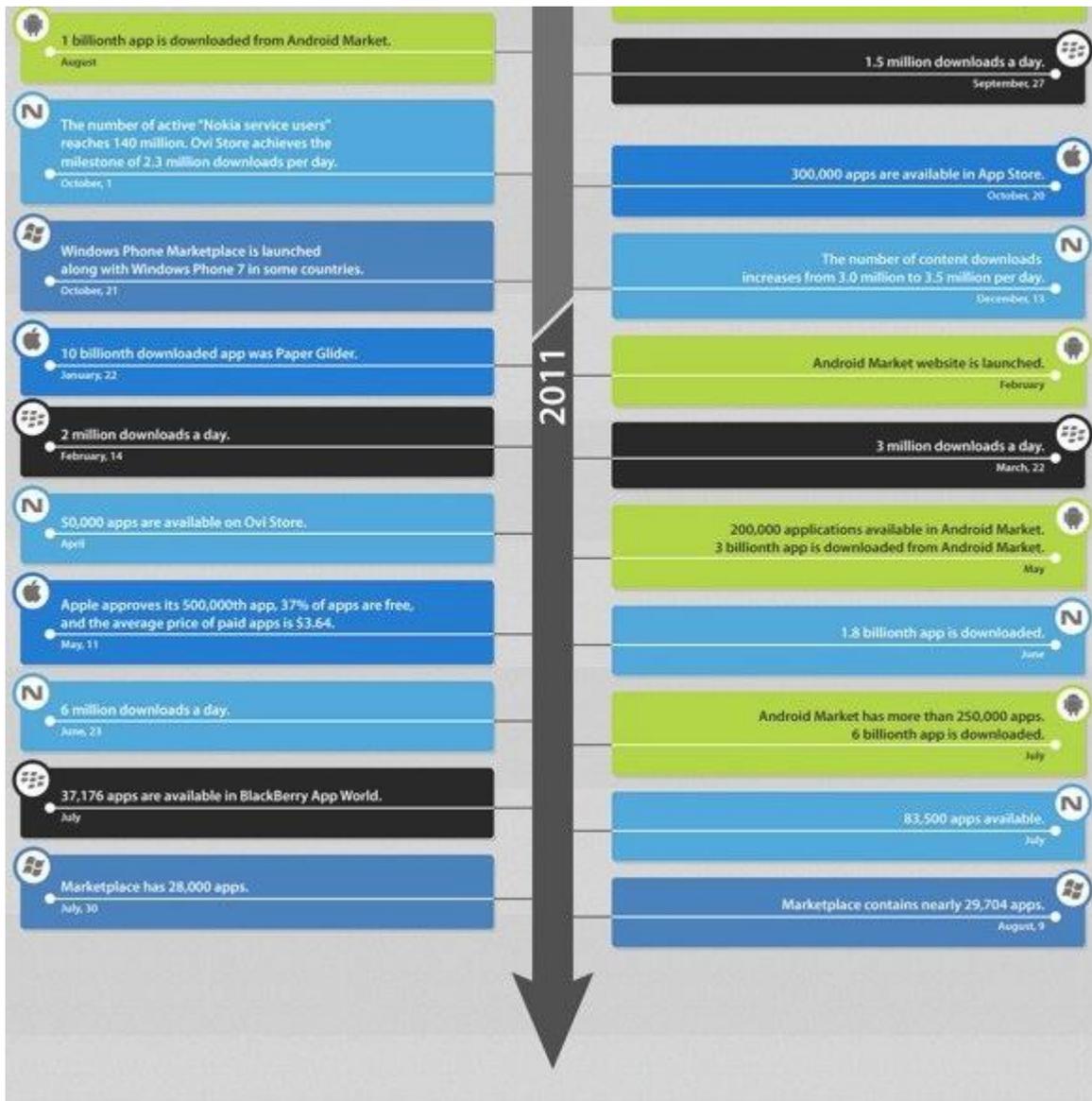


Imagen 4 - Infografía Reseña Historia P3

RIM (Research in Motion) en el mismo mes que el Android Market anunció también su BlackBerry App World. Ovi Store de Nokia se inauguró en 2009, comenzando una efímera carrera como número dos en tienda de aplicaciones detrás de Apple. El Marketplace de Windows Phone fue lanzado a finales de octubre del 2012. En julio de 2011 tenía cerca de 30.000 y actualmente cuenta con casi 50.000.

El Android Market alcanzó las 200.000 aplicaciones a principios de 2011 y casi ha duplicado su producción durante el año anterior. Cerrando el 2011 con 10 mil millones de descargas. Las cifras ahora son de 400.000 aplicaciones para el Android Market y 550.000 para la tienda de iOS. Sin duda, Apple App Store y

Android Market se encuentran en constante crecimiento, mientras que la Ovi Store de Nokia pasa por un momento de declive y BlackBerry sigue de forma fluida, aunque está a la espera de nuevos dispositivos y tratando de impulsar a los desarrolladores para que creen aplicaciones de nuevo para la plataforma (PhoneArena, 2012).

2.2.3. Población Objeto de Estudio

La población objeto de estudio serán los estudiantes de la universidad católica de Pereira en todas sus facultades.

2.2.4. Misión

La Universidad Católica de Pereira es una institución de Educación Superior inspirada en los principios de la fe católica, que asume con compromiso y decisión su función de ser apoyo para la formación humana, ética y profesional de los miembros de la comunidad universitaria y mediante ellos, de la sociedad en general.

La Universidad existe para el servicio de la sociedad y de la comunidad universitaria. El servicio a los más necesitados, es una opción fundamental de la institución, la cual cumple formando una persona comprometida con la sociedad, investigando los problemas de la región y comprometiéndose interinstitucionalmente en su solución. Es así como se entiende su carácter de Popular.

Guiada por los principios del amor y la búsqueda de la verdad y del bien, promueve la discusión amplia y rigurosa de las ideas y posibilita el encuentro de diferentes disciplinas y opiniones. En este contexto, promueve el diálogo riguroso y constructivo entre la fe y la razón. Como institución educativa actúa en los campos de la ciencia, la tecnología, el arte y la cultura, mediante la formación, la investigación y la extensión.

Inspirada en la visión del hombre de Jesús de Nazaret, posibilita la formación humana de sus miembros en todas las dimensiones de la existencia, generando una dinámica de auto superación permanente, asumida con autonomía y libertad, en un ambiente de participación y de exaltación de la dignidad humana.

La Universidad se propone hacer de la actividad docente un proyecto de vida estimulante orientado a crear y consolidar una relación de comunicación y de participación para la búsqueda conjunta del conocimiento y la formación integral.

Por tanto, a través de los programas de investigación se propone contribuir al desarrollo del saber y en particular al conocimiento de la región.

Mediante los programas de extensión se proyecta a la comunidad para contribuir al desarrollo, el bienestar y el mejoramiento de la calidad de vida. Para el logro de la excelencia académica y el cumplimiento de sus responsabilidades con la comunidad, la universidad fomenta programas de desarrollo docente y administrativo y propicia las condiciones para que sus miembros se apropien de los principios que la inspiran, bajo el compromiso de “Ser apoyo para llegar a ser gente, gente de bien y profesionalmente capaz”. (UCP, 2016)

2.2.5. Visión

La Universidad inspirada por los principios y valores cristianos será líder en los procesos de construcción y apropiación del conocimiento y en los procesos de formación humana, ética y profesional de sus estudiantes, de todos los miembros de la comunidad universitaria y de la sociedad. Será un escenario permanente para el diálogo riguroso y constructivo de la fe con la razón, en el contexto de la evangelización de la cultura y la inculturación del Evangelio.

Será reconocida por su capacidad para actuar como agente dinamizador del cambio y promover en la comunidad y en la familia sistemas armónicos de convivencia. La Universidad tendrá un claro sentido institucional de servicio orientado hacia sus estudiantes, profesores, personal administrativo y la comunidad.

Ejercerá liderazgo en programas y procesos de integración con la comunidad, los sectores populares, las empresas y el gobierno para contribuir al desarrollo sostenible. Se caracterizará por conformar un ambiente laboral y académico que sea expresión y testimonio de los principios y valores institucionales.

La Universidad tendrá la capacidad investigativa que le permita ser la institución con mayor conocimiento sobre los asuntos regionales. Consecuente con la realidad actual de un mundo interdependiente e intercomunicado, la Universidad fortalecerá sus vínculos con instituciones de su misma naturaleza tanto de orden nacional como internacional, y con otras instituciones.

La Universidad promoverá una reflexión pedagógica permanente en un ambiente de apertura para enseñar y aprender, dar y recibir en orden a la calidad y el servicio. (UCP, 2016)

2.3. MARCO CONCEPTUAL

2.3.1. Ingeniería de Software

Se conoce a la ingeniería de software como una disciplina que ayuda con el desarrollo de software de calidad, agilizando el desarrollo, mejorando la mantenibilidad y reduciendo errores, bugs entre otros. Con esta se pretende realizar un proceso de desarrollo basado en la aplicación de métodos, herramientas y teorías logrando con esto la producción de software de excelente calidad.

En general, los ingenieros de software adoptan un enfoque sistemático y organizado en su trabajo, ya que es la forma de producir software de alta calidad. Sin embargo, aunque la ingeniería consiste en seleccionar el método más apropiado para un conjunto de circunstancias, un enfoque más informal y creativo de desarrollo podría ser efectivo en algunas circunstancias. El desarrollo informal es apropiado para el desarrollo de sistemas basados web, los cuales requieren una mezcla de técnicas de software y diseño gráfico (Ian Sommerville Pearson, 2005).

2.3.2. Programación Web

En la programación de la web se pretende lograr que su desarrollo el cual es del lado del servidor brinde un servicio más especializado y personalizado al usuario logrando todo esto a través de la programación web, dado que esta tiene la misma importancia que el diseño web, estos dos se complementan y hacen que la eficiencia del sitio se ha mas importante. Por ello se empiezan a desarrollar alternativas a los CGI para solucionar este grave problema de rendimiento. Las soluciones vienen principalmente por dos vías. Por un lado se diseñan sistemas de ejecución de módulos más integrados con el servidor, que evitan que éste tenga que instanciar y ejecutar multitud de programas. La otra vía consiste en dotar al servidor de un intérprete de algún lenguaje de programación (RXML, PHP, VBScript, etc.) que nos permita incluir las páginas en el código de manera que el servidor sea quien lo ejecute, reduciendo así el tiempo de respuesta (Mateu, 2004).

2.3.3. Paradigma de la Programación Orientada a Objetos

La POO es un paradigma de programación (o técnica de programación) que utiliza objetos e interacciones en el diseño de un sistema.

Elementos de la POO: La POO está compuesta por una serie de elementos que se detallan a continuación.

Clase: Una clase es un modelo que se utiliza para crear objetos que comparten un mismo comportamiento, estado e identidad.

```
class Persona {  
  # Propiedades  
  # Métodos  
}
```

Objeto: Es una entidad provista de métodos o mensajes a los cuales responde (comportamiento); atributos con valores concretos (estado); y propiedades (identidad).

```
$persona = new Persona();  
/*  
El objeto, ahora, es $persona,  
que se ha creado siguiendo el modelo de la clase Persona  
*/
```

Método: Es el algoritmo asociado a un objeto que indica la capacidad de lo que éste puede hacer.

```
function caminar() {  
  #...  
}
```

Evento y Mensaje: Un evento es un suceso en el sistema mientras que un mensaje es la comunicación del suceso dirigida al objeto.

Propiedades y atributos:

Las propiedades y atributos, son variables que contienen datos asociados a un objeto.

```
$nombre = 'Juan';  
$edad = '25 años';  
$altura = '1,75 mts';
```

Características conceptuales de la POO: La POO debe guardar ciertas características que la identifican y diferencian de otros paradigmas de programación. Dichas características se describen a continuación.

Abstracción: Aislación de un elemento de su contexto. Define las características esenciales de un objeto.

Encapsulamiento: Reúne al mismo nivel de abstracción, a todos los elementos que

puedan considerarse pertenecientes a una misma entidad.

Modularidad: Característica que permite dividir una aplicación en varias partes más pequeñas (denominadas módulos), independientes unas de otras.

Ocultación (aislamiento): Los objetos están aislados del exterior, protegiendo a sus propiedades para no ser modificadas por aquellos que no tengan derecho a acceder a las mismas.

Polimorfismo: Es la capacidad que da a diferentes objetos, la posibilidad de contar con métodos, propiedades y atributos de igual nombre, sin que los de un objeto interfieran con el de otro.

Herencia: Es la relación existente entre dos o más clases, donde una es la principal (madre) y otras son secundarias y dependen (heredan) de ellas (clases "hijas"), donde a la vez, los objetos heredan las características de los objetos de los cuales heredan.

Recolección de basura: Es la técnica que consiste en destruir aquellos objetos cuando ya no son necesarios, liberándolos de la memoria (Bahit, duea, 2016)

2.3.4. FrameWork Bootstrap

Bootstrap es un framework de aplicaciones para usuario elegante e intuitivo, y de gran alcance para el desarrollo web más rápido y más fácil, creado por la marca Otto y Jacob Thorntony mantenido por el equipo central con el masivo apoyo y la participación de la comunidad (Cochran, 2012).

Es un framework completo, es decir, que tiene la rejilla de estilos CSS, componentes (html+css) y añade además una capa de JAVA para elementos de interacción.

Filosofía Mobile First: En la última versión se ha replanteado el framework para que tenga de forma más notoria la filosofía Mobile First, poniendo primero el foco en el desarrollo en dispositivos móviles.

También se han centrado en el Responsive Design para que se pueda visualizar en diferentes tamaños de pantalla (TaperBlog, 2015).

2.3.5. Base De Datos

En un sistema de bases de datos se encarga de que todos los archivos interrelacionados, y un conjunto de programas le den privilegios a los usuarios para poder consultar, modificar, eliminar e insertar información, en este sistema

maneja de forma abstracta la información respecto a cómo se almacena y se conserva los datos en un sistema de información, utilizando un sistema de estructura de modelo de datos y modelo entidad relación. En los Estados Unidos, muchas empresas utilizan el número de la seguridad social de una persona (un número único que el Gobierno de los Estados Unidos asigna a cada persona en los Estados Unidos) como identificador de cliente. Una relación es una asociación entre varias entidades. Por ejemplo, una relación impositor asocia un cliente con cada cuenta que tiene. El conjunto de todas las entidades del mismo tipo, y el conjunto de todas las relaciones del mismo tipo, se denominan respectivamente conjunto de entidades y conjunto de relaciones (Abraham Silberschatz, 2002).

2.3.6. FrameWork .Net

El FrameWork de .Net funciona bajo Windows y sirve para la ejecución de aplicaciones creadas bajo este mismo FrameWork. Actualmente el FrameWork se encuentra en la versión 4.6

.NET Framework es un entorno de ejecución administrado que proporciona diversos servicios a las aplicaciones en ejecución. Consta de dos componentes principales: Common Language Runtime (CLR), que es el motor de ejecución que controla las aplicaciones en ejecución, y la biblioteca de clases de .NET Framework, que proporciona una biblioteca de código probado y reutilizable al que pueden llamar los desarrolladores desde sus propias aplicaciones. (Microsoft, 2016)

2.3.7. Programación Por Capas

Se conoce como programación por capas a la arquitectura cliente-servidor en la cual se encarga de separar de la forma lógica de negocios de la diseño, la ventaja principal es que su desarrollo se genere en varios niveles, logrando con ello que si hay un cambio este puede corregido de acuerdo al nivel afectado, logrando con todo lo anterior la distribución de trabajo para la creación de la aplicación por niveles solo con conocer API que existe entre niveles.



Imagen 5 - Programación por capas

En el diseño de sistemas informáticos actual se suelen usar las arquitecturas multinivel o Programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El más utilizado actualmente es el diseño en tres niveles (o en tres capas) (Elvisor, 2015).

2.3.8. Metodología Ágil Mobile-D

Esta es una metodología ágil para el desarrollo de aplicaciones en la cual se pretende lograr varios ciclos de desarrollo de una manera rápida con muy pocas personas, esta consta de cinco etapas las cuales son: exploración, iniciación, producción, estabilización y prueba del sistema, en el desarrollo de cada una de estas se debe establecer las tareas y prácticas asociadas.

En la primera fase, Explorar, el equipo de desarrollo debe generar un plan y establecer las características del proyecto. Esto se realiza en tres etapas: establecimiento actores, definición del alcance y el establecimiento de proyectos. Las tareas asociadas a esta fase incluyen el establecimiento del cliente (los clientes que toman parte activa en el proceso de desarrollo), la planificación inicial del proyecto y los requisitos de recogida, y el establecimiento de procesos.

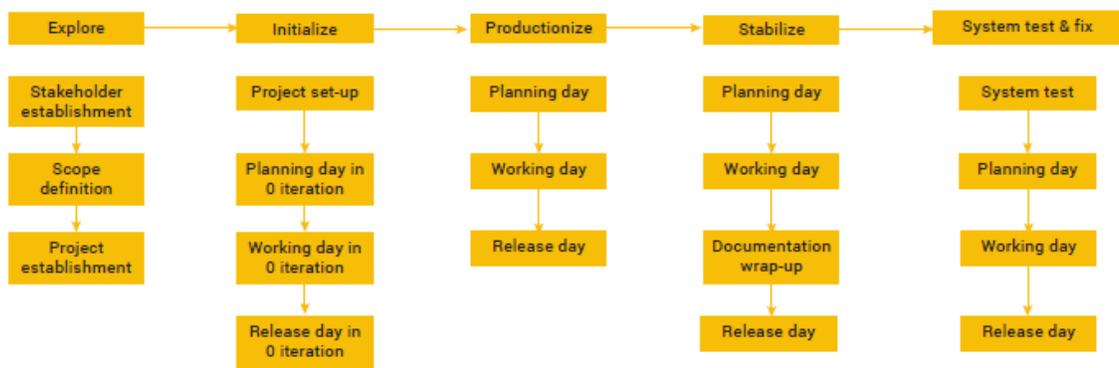


Imagen 6 - Metodología D-Mobile

En la siguiente fase, iniciación, los desarrolladores preparan e identifican todos los recursos necesarios. Se preparan los planes para las siguientes fases y se establece el entorno técnico como los recursos físicos, tecnológicos y de comunicaciones (incluyendo el entrenamiento del equipo de desarrollo). Esta fase

se divide en cuatro etapas: la puesta en marcha del proyecto, la planificación inicial, el día de prueba y día de salida.

En la fase de producción se repite la programación de tres días (planificación, trabajo, liberación) se repite iterativamente hasta implementar todas las funcionalidades. Primero se planifica la iteración de trabajo en términos de requisitos y tareas a realizar. Se preparan las pruebas de la iteración de antemano. Las tareas se llevarán a cabo durante el día de trabajo, desarrollando e integrando el código con los repositorios existentes. Durante el último día se lleva a cabo la integración del sistema (en caso de que estuvieran trabajando varios equipos de forma independiente) seguida de las pruebas de aceptación.

En la fase de estabilización, se llevan a cabo las últimas acciones de integración para asegurar que el sistema completo funciona correctamente. Esta será la fase más importante en los proyecto multi-equipo con diferentes subsistemas desarrollados por equipos distintos. En esta fase, los desarrolladores realizarán tareas similares a las que debían desplegar en la fase de “producción”, aunque en este caso todo el esfuerzo se dirige a la integración del sistema. Adicionalmente se puede considerar en esta fase la producción de documentación.

La última fase (prueba y reparación del sistema) tiene como meta la disponibilidad de una versión estable y plenamente funcional del sistema. El producto terminado e integrado se prueba con los requisitos de cliente y se eliminan todos los defectos encontrados. (Agudelo, 2011)

2.3.9. AndroidStudio

Es un lenguaje de programación para el desarrollo de aplicaciones de Android Studio, basado en IntelliJ IDEA este a su vez se convierte en un fácil programa para el desarrollo a su vez contiene una diversidad de herramientas para su mejor manejo y agilidad en los procesos. En la parte superior de potentes herramientas de edición de código y desarrolladores de IntelliJ, Android Studio ofrece aún más características que mejoran su productividad en la construcción de aplicaciones de Android, tales como: Un sistema de construcción basado en Gradle flexibles Construir variantes y generación de archivos APK múltiples plantillas de código para ayudarle a construir características de la aplicación comunes Un rico editor de diseño con soporte para la edición de arrastrar y soltar el tema herramientas para atrapar pelusa rendimiento, facilidad de uso, compatibilidad de versiones, y otros problemas Código encoge con ProGuard y recursos cada vez menor con Gradle El soporte integrado para Google Cloud Platform, por lo que es fácil de integrar Google mensajería en la nube y App Engine Esta página proporciona una introducción a las funciones básicas de Android Studio. Para las guías más detalladas para el uso de Android de estudio, comenzar a navegar por las páginas de la sección de flujo de trabajo. (developer.android, 2016)

2.3.10. Microsoft VisualStudio

Es un entorno de desarrollo que se encarga de la creación de sitios y aplicaciones web, estas pueden comunicarse entre diversas estaciones como lo son dispositivos móviles, consolas entre otras, esta a su vez soporta diversos lenguajes de programación. Visual Studio es un conjunto completo de herramientas de desarrollo para la generación de aplicaciones web ASP.NET, Servicios Web XML, aplicaciones de escritorio y aplicaciones móviles. Visual Basic, Visual C# y Visual C++ utilizan todos el mismo entorno de desarrollo integrado (IDE), que habilita el uso compartido de herramientas y hace más sencilla la creación de soluciones en varios lenguajes. Asimismo, dichos lenguajes utilizan las funciones de .NET Framework, las cuales ofrecen acceso a tecnologías clave para simplificar el desarrollo de aplicaciones web ASP y Servicios Web XML. (Microsoft, msdn.microsoft, 2007)

2.3.11. SQLITE

Se conoce a SQLite como una biblioteca la cual se encarga de la implementación de procesos autónomos sin la necesidad de contar con un servidor o una configuración, el dominio de SQLITE es público de esta maneja se puede tener un propósito tanto comercial o privado. SQLite es un motor de base de datos SQL incorporado. A diferencia de la mayoría de las otras bases de datos SQL, SQLite no tiene un proceso de servidor independiente. SQLite lee y escribe directamente en archivos de disco ordinarios. Una base de datos completa de SQL con varias tablas, índices, activadores y vistas, está contenida en un solo archivo de disco. La base de datos de formato de archivo es multiplataforma - se puede copiar libremente una base de datos entre sistemas de 32 bits y de 64 bits o entre big-endian y little-endian arquitecturas. Estas características hacen que SQLite una opción popular como un formato de archivo de la aplicación. (SQLite, 2016)

2.3.12. SDK

SDK responde a las siglas Software Development Kit, lo que viene a ser un kit de desarrollo de software. Con él podremos desarrollar aplicaciones y ejecutar un emulador del sistema Android de la versión que sea. Todas las aplicaciones Android se desarrollan en lenguaje Java con este kit. (Android-sdk, 2016)

2.3.13. Microsoft SQL SERVER

SQL Server es un sistema de gestión de bases de datos relacionales (RDBMS) de Microsoft que está diseñado para el entorno empresarial. SQL Server se ejecuta en T-SQL (Transact -SQL), un conjunto de extensiones de programación de Sybase y Microsoft que añaden varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas. (searchdatacenter, 2015)

2.3.14. Oracle

Es una base de datos relacional fabricada por Oracle Corporation. Oracle es básicamente un herramienta cliente/servidor para la gestión de base de datos la gran potencia que tiene y su elevado precio hace que solo se vea en empresas muy grandes y multinacionales, por norma general. Oracle Corporation: es una de las mayores compañías de software del mundo. Sus productos van desde bases de datos (Oracle) hasta sistemas de gestión. Cuenta además, con herramientas propias de desarrollo para realizar potentes aplicaciones, como Oracle Designer. (Oracle, 2015)

2.3.15. Normalización

Se pretende con la normalización la reducción de problemas respecto a la coherencia de datos, logrando una mejor simplificación de ellos además se busca conseguir mejorar comprensión, precisión, predicción y manejabilidad, esta a su vez se ha considerado de una forma no prioritaria, pero la teoría de la normalización está orientada a una aplicación práctica para el diseño de base de datos, logrando con ello evitar redundancia y Protección de la integridad de los datos.

Además, una base de datos normalizada reduce tiempos de ejecución, consigue una mejor indexación de las tablas e incluso mayor seguridad. Para conocer el proceso que hemos de seguir para normalizar una base de datos podemos ayudarnos de la orientación que nos brindan los libros que tratan el tema de forma monográfica o dentro de un enfoque más amplio, según sean nuestras necesidades. (powerdata, 2015)

2.3.16. XML

XML son las siglas del Lenguaje de Etiquetado Extensible. La expresión se forma a partir del acrónimo de la expresión inglesa eXtensible Markup Language. Se trata también de un lenguaje estándar que posee una Recomendación del World Wide Web Consortium: Extensible Markup Languages (XML) (<http://www.w3.org/TR/REC-xml/>). Con la palabra "Extensible" se alude a la no limitación en el número de etiquetas, ya que permite crear aquellas que sean necesarias. (coverpages, 2015)

2.3.17. JAVA

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como WORA, o "write once, run anywhere"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es, a partir de 2012, uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente-servidor de web, con unos 10 millones de usuarios reportados. (James Gosling, 2015)

2.3.18. C#

C# es un lenguaje de programación que se ha diseñado para compilar diversas aplicaciones que se ejecutan en .NET Framework. C# es simple, eficaz, con seguridad de tipos y orientado a objetos. Las numerosas innovaciones de C# permiten desarrollar aplicaciones rápidamente y mantener la expresividad y elegancia de los lenguajes de estilo de C.

Visual C# es una implementación del lenguaje C# de Microsoft. Visual Studio ofrece compatibilidad con Visual C# con un completo editor de código, un compilador, plantillas de proyecto, diseñadores, asistentes para código, un depurador eficaz y de fácil uso y otras herramientas. La biblioteca de clases de .NET Framework ofrece acceso a numerosos servicios de sistema operativo y a otras clases útiles y adecuadamente diseñadas que aceleran el ciclo de desarrollo de manera significativa. (Microsoft, Msdn.Microsoft, 2016)

2.3.19. Webservice

Se conoce como web service al conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Con esto se logra que diversas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como internet.

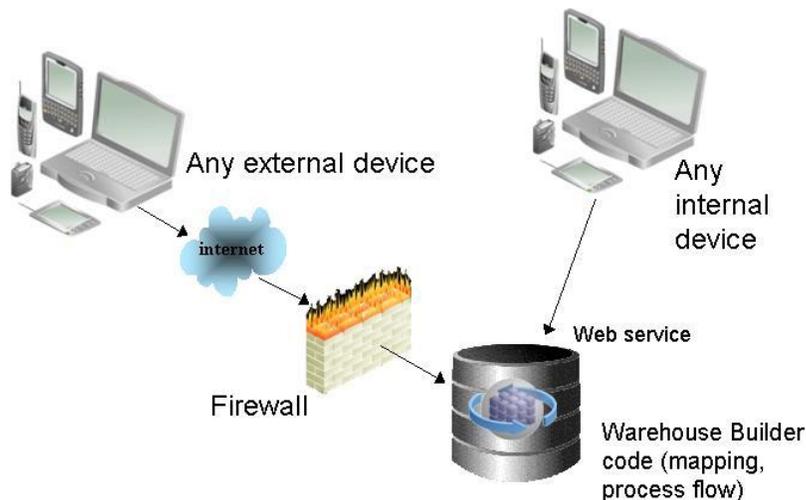


Imagen 7 – Funcionamiento Webservice

De una manera más clara se podría decir que un web service es una función que diferentes servicios o equipos utilizan; es decir, solo se envían parámetros al servidor (lugar donde está alojado el web service) y éste responderá la petición. Entre algunas que se manejan de utilizar servicios webs en las aplicaciones destacan las siguientes:

- Aportan interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen.
- Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- Al apoyarse en HTTP, los servicios Web pueden aprovecharse de los sistemas de seguridad firewall sin necesidad de cambiar las reglas de filtrado.
- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.

- Permiten la interoperabilidad entre plataformas de distintos fabricantes por medio de protocolos estándar y abiertos. Las especificaciones son gestionadas por una organización abierta, la W3C, por tanto no hay secretismos por intereses particulares de fabricantes concretos y se garantiza la plena interoperabilidad entre aplicaciones.

La principal ventaja de utilizar un servicio web es que son bastante prácticos debido a que son independientes de las aplicaciones. (culturacion, 2012)

2.3.20. Seguridad WebService

En la seguridad WebService se pretende asegurar a SOAP, se requiere que el mensaje que está transportando este se ha asegurado el cual será mantenido en secreto ante algunos receptores no deseados y conocer si el cliente está autorizado para recibir el mensaje, este se empaqueta a través de XML, teniendo en cuenta los diversos roles respecto a la encriptación y desencriptación de contenidos del mensajes, validación firmas digitales, implementación protocolos de seguridad extendidos, en la ubicación del web service se utiliza un mecanismo de autenticación para evitar que este se ha mal usado, como la protección del documento WSDL mediante SSL solicitando la autenticación del cliente en caso de que desee accederlo. Una vez entendida la problemática respecto a la seguridad de la información se puede entrar en detalle en los conceptos y mecanismos existentes para brindar seguridad en XML y extenderlos a los web services en general. Los tres mecanismos básicos son XML Signature, XML Encryption y SAML, que serán explicados a lo largo del presente documento, al igual que sus extensiones a SOAP en el marco de WS-Security. (Zunino, 2006)

2.3.21. Protocolo HTTPS

Este protocolo se encarga de asegurar el protocolo de transferencia de hipertexto, con el cual a su vez mueve información en la World Wide Web. Básicamente, conectarse a un website vía “HTTPS” (en lugar de conectarse a través del conocido “HTTP”) significa que cualquier dato que intercambies con dicho sitio pasará por encima de la seguridad de la capa de transporte encriptada (TLS) o su predecesora, la capa de conexión segura (SSL).

Las conexiones “HTTPS” se encuentran señalizadas con un pequeño ícono de un candado en algún lugar de tu barra de direcciones, o en la “s” luego del “http” en la URL. “HTTPS” significa que cualquier dato que pases por un website estará encriptado mientras se mueva. (Kaspersky, 2013)

2.3.22. Servidor Web (Internet Information Service)

El rol de servidor web (IIS) incluye Internet Information Services (IIS) 7, que es una plataforma web unificada que integra IIS, ASP.NET, Windows Communication Foundation y Windows SharePoint Services. IIS 7 permite compartir información con usuarios en Internet, en una intranet o en una extranet. Windows Server® 2008 ofrece IIS 7.0, que también se incluye con algunas ediciones de Windows Vista®. Windows Server® 2008 R2 ofrece IIS 7,5, que también se incluye en algunas ediciones de Windows® 7. (Microsoft, 2008)

2.3.23. GIT

El directorio de Git es donde Git almacena los metadatos y la base de datos de objetos para tu proyecto. Es la parte más importante de Git, y es lo que se copia cuando clonas un repositorio desde otro ordenador.

El directorio de trabajo es una copia de una versión del proyecto. Estos archivos se sacan de la base de datos comprimida en el directorio de Git, y se colocan en disco para que los puedas usar o modificar.

El área de preparación es un sencillo archivo, generalmente contenido en tu directorio de Git, que almacena información acerca de lo que va a ir en tu próxima confirmación. A veces se le denomina índice, pero se está convirtiendo en estándar el referirse a ella como el área de preparación. (GIT, 2016)

2.3.24. Team Foundation Server

Este se encarga de brindar funciones de control de código fuente, este a su vez proporciona un sitio web del portal de proyecto de equipo buscando la creación de informes y administración de proyectos, se almacena la información de los datos de seguimiento de elementos de trabajo, el control de código fuente, las compilaciones y las herramientas de pruebas. Todos los proyectos de equipo se almacenan y se administran con Team Foundation Server (TFS). Para empezar a trabajar en un proyecto de equipo, primero deberá conectarse al servidor de TFS adecuado. Antes de intentar conectarse a TFS compruebe que es miembro de algún grupo del servidor de TFS y que dispone de los permisos necesarios, o que su cuenta se ha agregado directamente al servidor de TFS y se ha configurado con los permisos correspondientes a la función que desempeña en el proyecto. Si no se ha agregado al servidor de TFS, bien como usuario o como miembro de un grupo, no podrá conectarse al servidor de TFS. (Microsoft, 2007)

2.3.25. Window Communication Foundation

Este se encarga de la creación de aplicaciones orientadas a servicios, es posible él envió de mensajes asíncronos de extremo de servicio a otro, los servicios pueden alojados en un extremo de un cliente que contenga un servicio hospedado con él se encarga del solicitar datos de uno de los extremos, los mensajes enviados suelen ser simples como un carácter o una palabra como XML, o tan complejos como los binarios

El servidor de aplicaciones características de hospedaje de Windows Server AppFabric se ha diseñado específicamente para implementar y administrar aplicaciones que utilizan WCF para las comunicaciones. Características de hospedaje incluye sofisticadas opciones de configuración y herramientas diseñadas específicamente para las aplicaciones habilitadas para WCF. (Microsoft, 2016)

2.3.26. Diagramas UML

El Lenguaje de Modelado Unificado (UML:Unified Modeling Language) es la sucesión de una serie de métodos de análisis y diseño orientadas a objetos que aparecen a fines de los 80's y principios de los 90s.UML es llamado un lenguaje de modelado, no un método. Los métodos consisten de ambos de un lenguaje de modelado y de un proceso.

El UML, fusiona los conceptos de la orientación a objetos aportados por Booch, OMT y OOSE (Booch, G. et al., 1999).

UML incrementa la capacidad de lo que se puede hacer con otros métodos de análisis y diseño orientados a objetos. Los autores de UML apuntaron también al modelado de sistemas distribuidos y concurrentes para asegurar que el lenguaje maneje adecuadamente estos dominios.

El lenguaje de modelado es la notación (principalmente gráfica) que usan los métodos para expresar un diseño. El proceso indica los pasos que se deben seguir para llegar a un diseño.

La estandarización de un lenguaje de modelado es invaluable, ya que es la parte principal del proceso de comunicación que requieren todos los agentes involucrados en un proyecto informático. Si se quiere discutir un diseño con alguien más, ambos deben conocer el lenguaje de modelado y no así el proceso que se siguió para obtenerlo. (Cornejo, 2008)

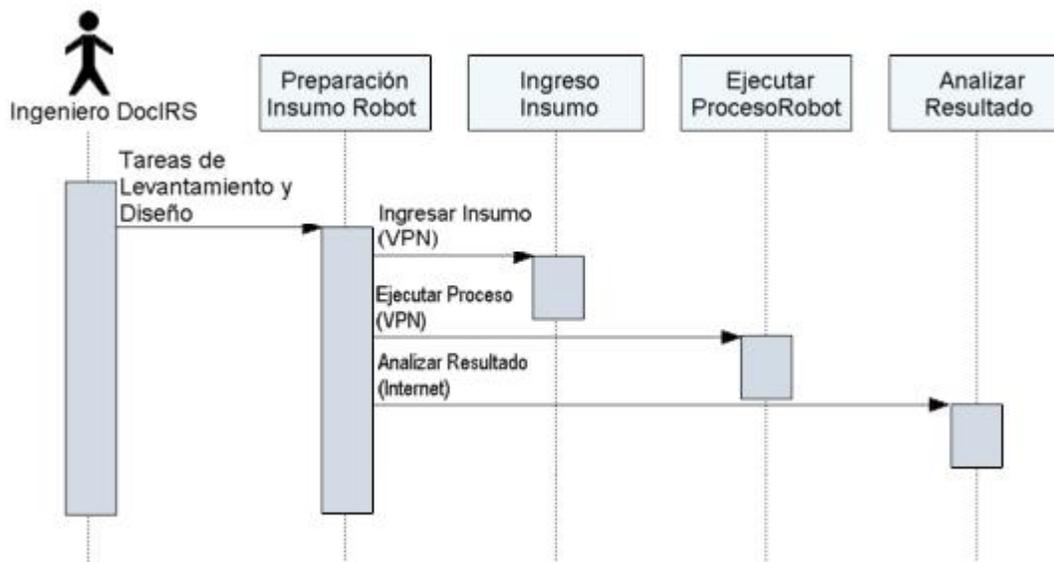


Imagen 8 - Características herramienta UML

2.3.27. HTML

HTML se encarga de la construcción de elementos más básico de una página web y se usa para crear y representar visualmente una página web. Este a su vez define su funcionalidad, añadiendo el Hiper Texto con la conexión a la web la cual puede ser creada y a su vez subida a internet logrando imágenes y también otro tipo de elementos multimedia. Con la ayuda de HTML todos pueden hacer sitios web estáticos y dinámicos. HTML es el lenguaje que describe la estructura y el contenido semántico de un documento web. El contenido dentro de una página web es etiquetado con elementos HTML como , <title>, <p>, <div>, y así sucesivamente. Estos elementos conforman los bloques de construcción de un sitio web. (AngelfQC, 2016)

2.3.28. ASP.NET

Este es un modelo de desarrollo Web unificado en el cual se incluyen aplicaciones Web empresariales, este a su vez forma parte de .NET Framework y al codificar las aplicaciones ASP.NET tiene acceso a las clases. A continuación, se pueden crear páginas de contenido individuales con el contenido específico de la página que se desee mostrar. Cuando los usuarios solicitan las páginas de contenido, las combinan con la página principal con el fin de generar un resultado

que combine el diseño de la página principal con el de la página de contenido. (Microsoft, 2007)

2.3.29. Web Service REST

REST significa Transferencia de estado representacional. Rest se utiliza para crear servicios Web que son de peso ligero, fácil de mantener, y escalable en la naturaleza. Un servicio que se basa en la arquitectura REST se llama un servicio web REST. El protocolo subyacente para REST es HTTP, que es el protocolo web básico. REST es una manera de acceder a los recursos que se encuentran en un entorno particular. Por ejemplo, usted podría tener un servidor que podría ser el anfitrión de importantes documentos o imágenes o videos. Todos estos son un ejemplo de los recursos. Si un cliente, por ejemplo un navegador web necesita cualquiera de estos recursos, tiene que enviar una solicitud al servidor para acceder a estos recursos. Ahora REST define una manera de cómo se puede acceder a estos recursos. (guru99, 2013)

2.3.30. Oracle 11G

Se encarga de la conversión de un servidor el cual se encarga de procesos a un servidor basado en threads, En UNIX, Oracle utiliza procesos para implementar tareas de segundo plano, como el generador de base de datos (DBW0), el generador de registros (LGWR), los distribuidores, servidores compartidos, entre otros. Asimismo, cada conexión dedicada, realizada en la base de datos provoca el arranque de otro proceso del sistema operativo por parte de esa sesión. En Windows, en cambio, todos estos procesos se implementan como threads dentro de un único y amplio proceso. Esto significa que para cada instancia de la base de datos Oracle, hay un solo proceso ejecutándose en Windows para el propio servidor de base de datos Oracle. (Oracle, 2007)

2.3.31. Librería Gson

Es una biblioteca del lenguaje de programación Java el cual se encarga de convertir objetos Java en su representación JSON. También se puede utilizar para convertir una cadena JSON a un objeto Java equivalente. Sin embargo, la mayoría de ellos requieren que coloque anotaciones Java en sus clases; algo que no se puede hacer si usted no tiene acceso al código fuente. La mayoría también

no apoyan totalmente el uso de genéricos de Java. Gson considera tanto de éstos como objetivos de diseño muy importantes. (google, 2015)

2.3.32. Librería DBFlow

DBFlow se construye a partir de una colección de las mejores características de muchas bibliotecas de bases de datos de la manera más eficiente posible. Además, se construye no sólo para hacer que sea mucho más fácil tratar con las bases de datos en Android, sino también para proporcionar extensibilidad. No deje que un ORM o biblioteca en su camino, dejar que el código que escriba en sus aplicaciones sea la mejor posible. (Raizlabs, 2015)

2.3.33. Librería Retrofit 2

Retrofit es un cliente REST para Android y Java, desarrollada por Square, Permite hacer peticiones GET, POST, PUT, PATCH, DELETE y HEAD; gestionar diferentes tipos de parámetros y pasar automáticamente la respuesta a un POJO (Plain Old Java Object). (square, 2014)

2.3.34. Librería Volley

Volley es una librería desarrollada por Google para optimizar el envío de peticiones Http desde las aplicaciones Android hacia servidores externos. Este componente actúa como una interfaz de alto nivel, liberando al programador de la administración de hilos y procesos tediosos de parsing, para permitir publicar fácilmente resultados en el hilo principal. (Revelo, 2015)

2.4. ESPECIFICACIÓN DE LA SOLUCIÓN

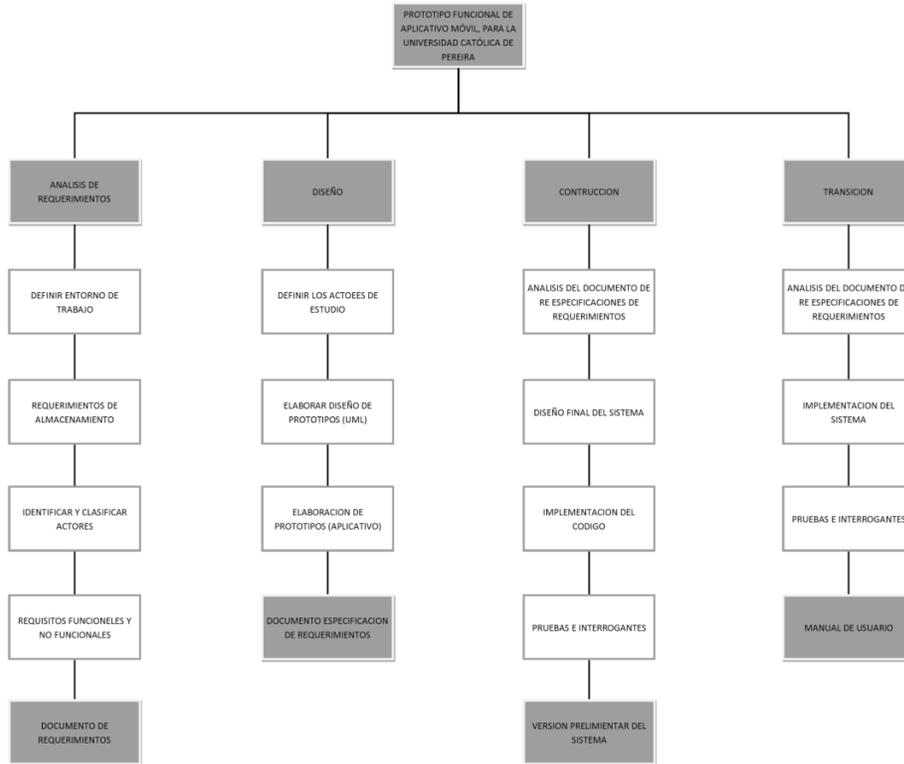


Imagen 9 - EDT Proyecto UCP-PIM

2.4.1. Requerimientos

- Requerimientos Funcionales

Registrar Dispositivo

(CU_REQ_01)

- El estudiante deberá estar activo en la universidad o ser egresado
- Vincular el dispositivo por medio de la cedula del estudiante y el IMEI del celular.

Configuración

(CU_REQ_02)

- Borrar la información almacenada temporalmente en el teléfono.
- Sincronizar la información.

(CU_REQ_03)

- Cambiar de idioma a inglés o español

(CU_REQ_04)

- Ver información personal almacenada del usuario actual del dispositivo.
- Cambiar información personal del estudiante.

(CU_REQ_05)

- Agregar o cambiar PIN o contraseña

Iniciar Sesión

(CU_REQ_06)

- Verificar el estado de la cuenta del estudiante, si ya está vinculada o no.
- En el primer log in después de verificar el estado del estudiante, deberá aceptar términos y uso del aplicativo móvil.

(CU_REQ_07)

- En el primer log in se re direccionará a agregar Contraseña o PIN
- Acceder por medio de PIN o Contraseña.

(CU_REQ_08)

- Con 3 intentos fallidos el aplicativo enviará un correo electrónico informando sobre actividad sospechosa.

Sistema

(CU_REQ_09)

- La aplicación será compatible desde la versión 4 de Android hasta la actual.
- El dispositivo Android deberá que contar con un IMEI.

(CU_REQ_10)

- La aplicación exigirá la activación del GPS al dispositivo móvil.
- Deberá estar conectado a red de datos o WI-FI al momento de vincular.
- Deberá estar conectado a red de datos o WI-FI al momento de sincronizar.

UCP

(CU_REQ_11)

- Consultar información de los horarios del estudiante.
- Consultar información de las calificaciones del estudiante
- Consultar información de los libros de la biblioteca. (SIABUC)

(CU_REQ_12)

- Contará con un pequeño formulario móvil y web donde se podrán publicar noticias.
- Ver noticias de la universidad y la comunidad de la misma.

(CU_REQ_13)

- Ver información de la facultad y de sus respectivos programas.
- Ver letra del himno de la universidad.

Ayuda

(CU_REQ_14)

- Ver un FAQ del aplicativo.
- Acceso directo para llamar a la Universidad.
- Autoría de los desarrolladores.

- Licencia del producto y términos de uso.
- Acerca del aplicativo.

- Requerimientos Funcionales
 - Mantener los datos almacenados y seguros.
 - Ser de fácil descarga y fácil instalación.
 - Ser fácil de analizar para una fácil mantenibilidad.
 - Las interfaces de los usuarios deben ser amigables e intuitivas.
 - Necesita al menos 50 megas para un óptimo funcionamiento.
 - Se identificará por los colores de la Universidad Católica de Pereira.
 - Mostrar mensajes de información entendibles y concretos.
 - Tendrá un estándar de codificación.
 - Estará publicada en la tienda de aplicaciones de Google Play.
 - Tendrá un tiempo de respuesta rápido al momento de sincronizar.

2.4.2. UML

- Diagrama de casos de uso
 - Actores

| | |
|-------------------------------------|---|
| Actor: | Estudiante |
| Nombres de los Casos de uso: | Vincular dispositivo,... |
| Tipo: | Primario |
| Descripción: | El estudiante es el actor principal del sistema ya que es el que interactuará más con el mismo. |
| ***** | |
| Actor: | WebService |
| Nombres de los Casos de uso: | Vincular dispositivo,... |

| | |
|-------------------------------------|--|
| Tipo: | Secundario |
| Descripción: | El webservice es el API que intercomunicará al dispositivo móvil con las bases de datos correspondientes al evento ejecutado por el actor Estudiante |
| ***** | |
| Actor: | Base de datos local |
| Nombres de los Casos de uso: | Todos |
| Tipo: | Secundario |
| Descripción: | La base de datos local permitirá al actor Estudiante acceder en cualquier momento a su información personal y en estado offline |
| ***** | |
| Actor: | Base de datos servidor |
| Nombres de los Casos de uso: | Vincular dispositivo,... |
| Tipo: | Secundario |
| Descripción: | De las base de datos del servidor se sincronizará el dispositivo cuando sea necesario o cuando el actor estudiante lo solicite. |

Tabla 1 – Actores

- Registrar Dispositivo (CU_REQ_01)

| Campo de Caso de uso | Descripción |
|-------------------------------|---|
| Nombre del caso de uso | Registrar Dispositivo (CU_REQ_01) |
| Actores | Estudiante, Webservice, Base de datos Servidor |
| Tipo | Primario |
| Proposito | Verificar y vincular dispositivos. |
| Resumen | Verificar y vincular un dispositivo a un estudiante para permitirle usar el aplicativo y pueda ver su información correspondiente. |
| Precondiciones | Se requiere la validación correcta del usuario |
| Flujo principal | Se presenta al usuario la Pantalla Principal. El Usuario puede seleccionar entre las siguientes opciones: "Vincular" y "Salir". Si la actividad seleccionada es "Vincular", se ejecuta el caso de uso Vincular dispositivo. Si la opción seleccionada es "Salir", el aplicativo se cerrará. |
| Sub flujo | Estar registrado en las bases de datos como estudiante activo. |
| Excepciones | Usuario no activo o no encontrado. |
| PostCondiciones | Mostrar mensaje de vinculación exitosa o usuario no encontrado o usuario no activo. |

Tabla 2 - Registrar Dispositivo (CU_REQ_01)

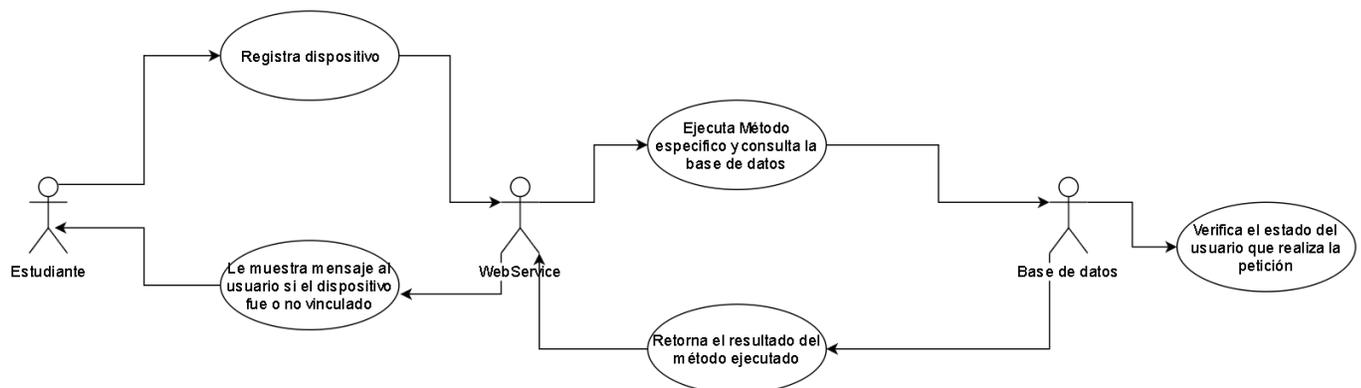


Imagen 10 - Registrar Dispositivo (CU_REQ_01)

- Borrar y Sincronizar (CU_REQ_02)

| Campo de Caso de uso | Descripción |
|-------------------------------|--|
| Nombre del caso de uso | Borrar y Sincronizar (CU_REQ_02) |
| Actores | Estudiante, WebService, Base de datos local, Base de datos Servidor. |
| Tipo | Primario |
| Propósito | Borrar y sincronizar información del servidor. |
| Resumen | Borrar la información almacenada temporalmente en el teléfono; Sincronizar la información |
| Precondiciones | Debe haber conexión a internet para sincronizar. |
| Flujo principal | Se presenta al usuario la Pantalla con varias opciones, entre ellas "Borrar información" y "Sincronizar". Si el usuario elige "Borrar información" el sistema procederá a borrar toda la información temporal almacenada en la base de datos local; Si el usuario elige "Sincronizar" el sistema procederá a descargar información de la base de datos del servidor y almacenarla en la base de datos local. |
| Sub flujo | Estar registrado en las bases de datos como estudiante activo; Tener cuenta vinculada; Tener sesión iniciada. |
| Excepciones | No se pueda conectar al servidor; Sin conexión a internet. |
| PostCondiciones | Mostrar mensaje de proceso exitoso o error al realizar el proceso. |

Tabla 3 - Borrar y Sincronizar (CU_REQ_02)

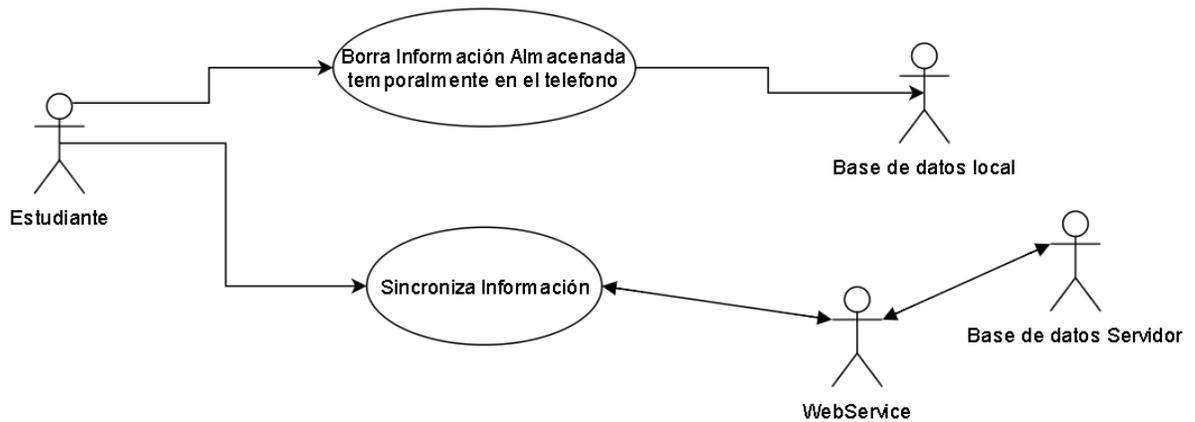


Imagen 11 - Borrar y Sincronizar (CU_REQ_02)

- Cambiar Idioma (CU_REQ_03)

| Campo de Caso de uso | Descripción |
|------------------------|--|
| Nombre del caso de uso | Cambiar Idioma (CU_REQ_03) |
| Actores | Estudiante, WebService, Base de datos local, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | Cambiar idioma del sistema |
| Resumen | Cambiar de idioma a inglés o español |
| Precondiciones | Ninguna |
| Flujo principal | Se presenta al usuario la pantalla con 2 opciones: "Español" o "Ingles"; Si el estudiante elige "Español", cambia a este idioma. Si el usuario elige "Ingles", cambia a este idioma. |
| Sub flujo | Estar registrado en las bases de datos como estudiante activo; Tener cuenta vinculada; Tener sesión iniciada. |
| Excepciones | Error al cambiar de idioma. |
| PostCondiciones | Mostrar mensaje de cambio de idioma exitoso; Mostrar mensaje de error al cambiar idioma. |

Tabla 4 - Cambiar Idioma (CU_REQ_03)

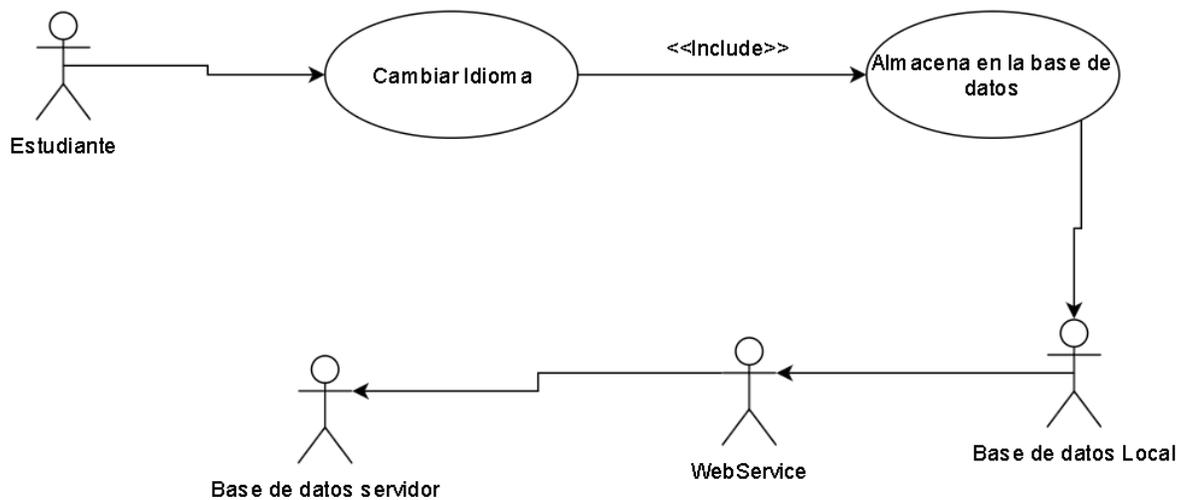


Imagen 12 - Cambiar Idioma (CU_REQ_03)

- Información Personal (CU_REQ_04)

| Campo de Caso de uso | Descripción |
|-------------------------------|---|
| Nombre del caso de uso | Información Personal (CU_REQ_04) |
| Actores | Estudiante, WebService, Base de datos local, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | Actualizar y ver información personal del estudiante. |
| Resumen | Ver información personal almacenada del usuario actual del dispositivo. Cambiar información personal del estudiante. |
| Precondiciones | Ninguna |
| Flujo principal | Se presenta al usuario una pantalla con 2 opciones: "Ver Información" y "Guardar"; Si se elige la opción "Ver Información", se abrirá una ventana con la información personal del estudiante. Si se elige la opción "Guardar" se almacenarán los datos en la base de datos. |
| Sub flujo | Estar registrado en las bases de datos como estudiante activo; Tener cuenta vinculada; Tener sesión iniciada. |
| Excepciones | Error al ver o guardar información |
| PostCondiciones | Mostrar mensaje de información guardada; Mostrar mensaje de error al guardar; Mostrar mensaje de error al consultar. |

Tabla 5 - Información Personal (CU_REQ_04)

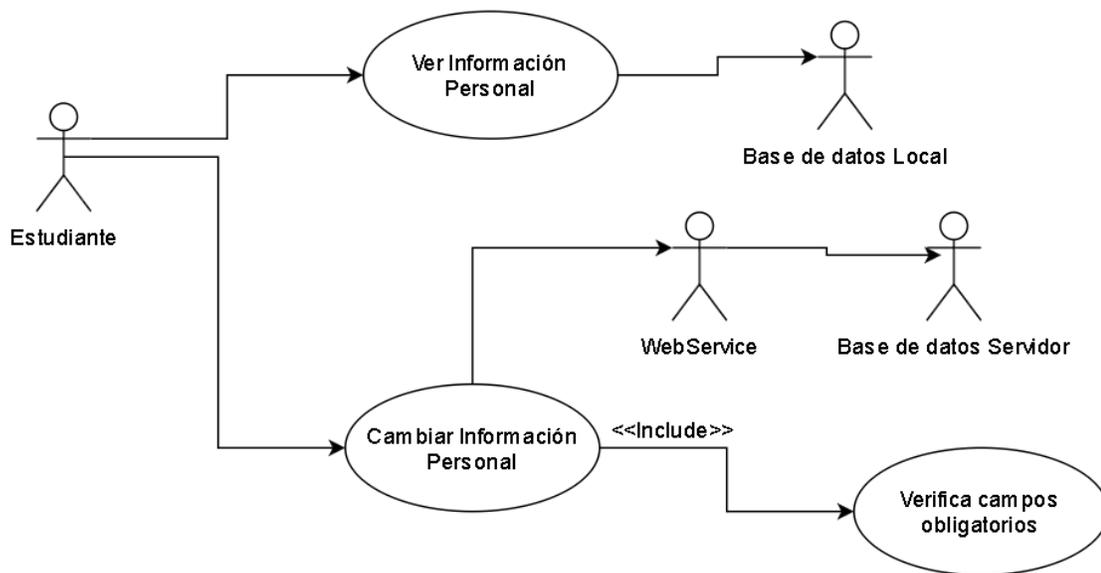


Imagen 13 - Información Personal (CU_REQ_04)

- PIN o Contraseña (CU_REQ_05)

| Campo de Caso de uso | Descripción |
|-------------------------------|---|
| Nombre del caso de uso | PIN o Contraseña (CU_REQ_05) |
| Actores | Estudiante, WebService, Base de datos local, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | Agregar o cambiar PIN o Contraseña |
| Resumen | Agregar o cambiar PIN o Contraseña |
| Precondiciones | Ninguna |
| Flujo principal | Se presenta al usuario una pantalla con 2 opciones: "PIN" y "Contraseña"; Si se elige la opción "PIN", se abrirá una ventana para que el usuario agregue o cambie su PIN. Si se elige la opción "Contraseña", Aparecerá una ventana para que el usuario cambie su contraseña. |
| Sub flujo | Estar registrado en las bases de datos como estudiante activo; Tener cuenta vinculada; Tener sesión iniciada. |
| Excepciones | Error al verificar PIN |
| PostCondiciones | Mostrar mensaje de error al cambiar PIN o Contraseña; Mostrar mensaje de error al verificar que el PIN o Contraseña Coincidan; Mostrar mensaje de cambio exitoso. |

Tabla 6 - PIN o Contraseña (CU_REQ_05)

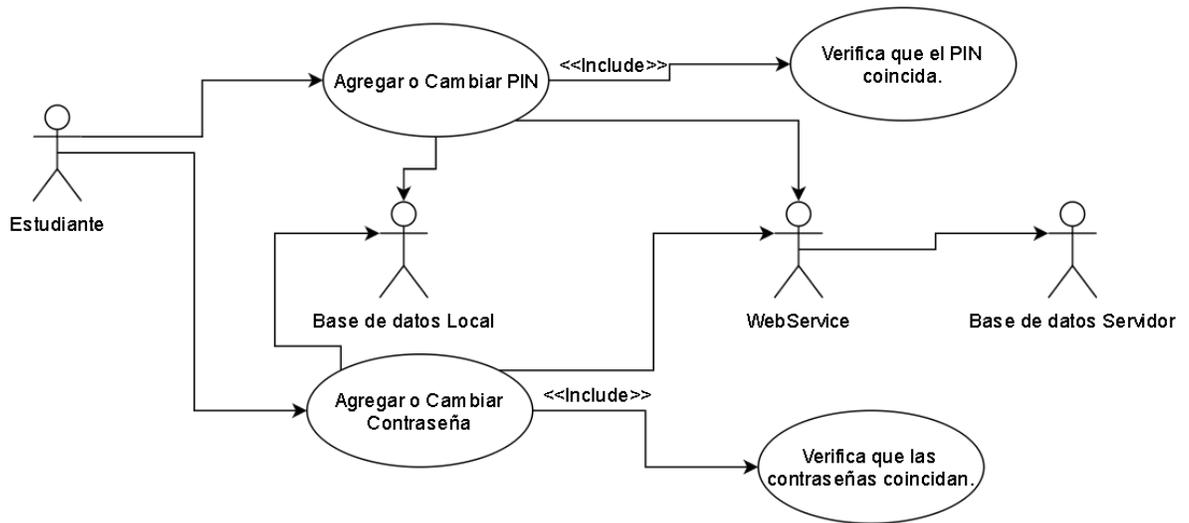


Imagen 14 - PIN o Contraseña (CU_REQ_05)

- Vincular cuenta estudiante (CU_REQ_06)

| Campo de Caso de uso | Descripción |
|------------------------|--|
| Nombre del caso de uso | Vincular cuenta estudiante (CU_REQ_06) |
| Actores | Estudiante, WebService, Base de datos Servidor. |
| Tipo | Primario |
| propósito | Vincular cuenta estudiante |
| Resumen | cuando el estudiante acceda vinculara su cuenta con la cual ya está registrado al momento de la vinculación saldrá un resumen de los términos y condiciones con respecto al uso de la plataforma |
| Precondiciones | Ninguna |
| Flujo principal | al momento de la vinculación el usuario conocerá las condiciones y restricciones que tendrá al momento del uso de la plataforma |
| Sub flujo | Estar conectado a internet para realizar el proceso |
| Excepciones | Error al momento de que la cuenta no se esté vinculada |
| PostCondiciones | Mostrar mensajes que incluyan que la cuenta que está accediendo no corresponde a una asociada o no se está dando la información correctamente |

Tabla 7 - Vincular cuenta estudiante (CU_REQ_06)

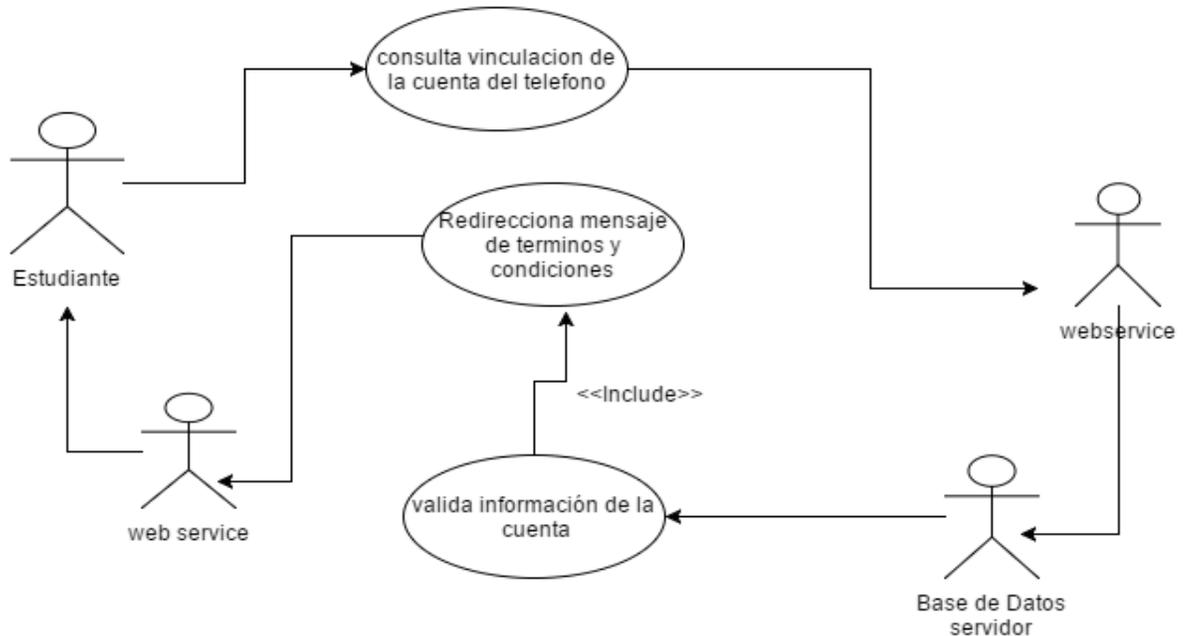


Imagen 15-Vincular cuenta estudiante (CU_REQ_06)

- Agregar o cambiar contraseña (CU_REQ_07)

| Campo de Caso de uso | Descripción |
|------------------------|---|
| Nombre del caso de uso | Agregar o cambiar contraseña (CU_REQ_07) |
| Actores | Estudiante, WebService, Base de datos Servidor. |
| Tipo | Primario |
| propósito | El estudiante pueda modificar su contraseña |
| Resumen | Cuando el estudiante acceda pueda cambiar su contraseña |
| Precondiciones | El estudiante deberá ingresar con la cuenta y contraseña asociada |
| Flujo principal | Al momento del ingreso a la aplicación el estudiante el estudiante podrá cambiar la contraseña y este se logueara |
| Sub flujo | Tener contraseña y cuenta correcta al momento del ingreso |
| Excepciones | Error al momento si la cuenta o contraseña son incorrectas |
| PostCondicioness | El estudiante deberá tener descargada la App y saber su cuenta y contraseña |

Tabla 8-Agregar o cambiar contraseña (CU_REQ_07)

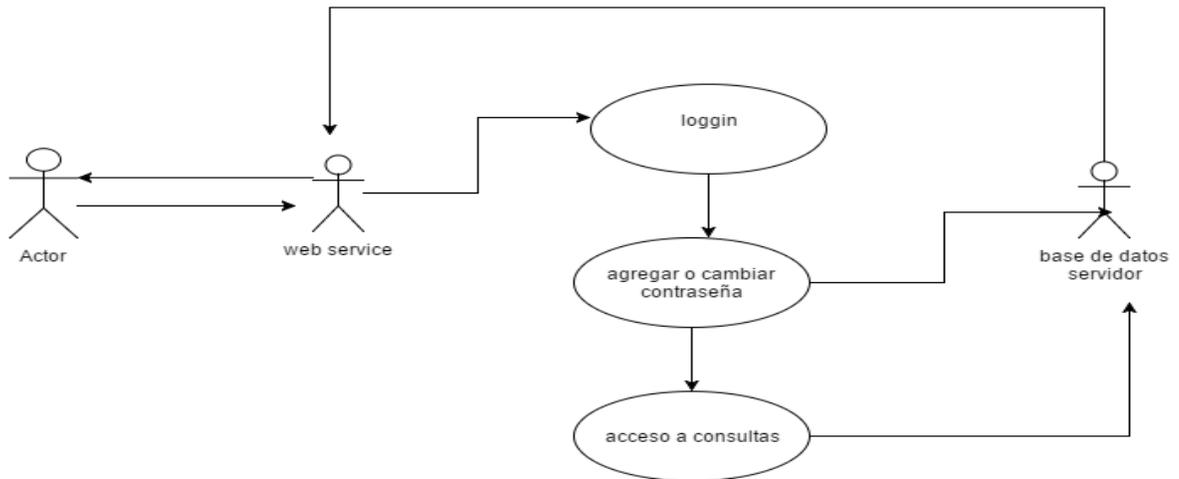


Imagen 16-Agregar o cambiar contraseña (CU_REQ_07)

- Contraseña invalida (CU_REQ_08)

| Campo de Caso de uso | Descripción |
|------------------------|--|
| Nombre del caso de uso | Contraseña invalida (CU_REQ_08) |
| Actores | Estudiante, WebService, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | validar la contraseña y su cuenta asociada |
| Resumen | si el usuario ingresa más de tres veces continuas una contraseña errada este remitirá un mensaje a su correo |
| Precondiciones | el estudiante deberá tener su cuenta y contraseña asociada |
| Flujo principal | al momento que el usuario ingresa a la aplicación deberá registrarse con su usuario y contraseña |
| Sub flujo | validar contraseña y cuenta asociada |
| Excepciones | si la cuenta está asociada con la contraseña eta ingresara |
| PostCondiciones | el estudiante deberá previamente saber su cuenta y contraseña |

Tabla 9-Contraseña invalida (CU_REQ_08)

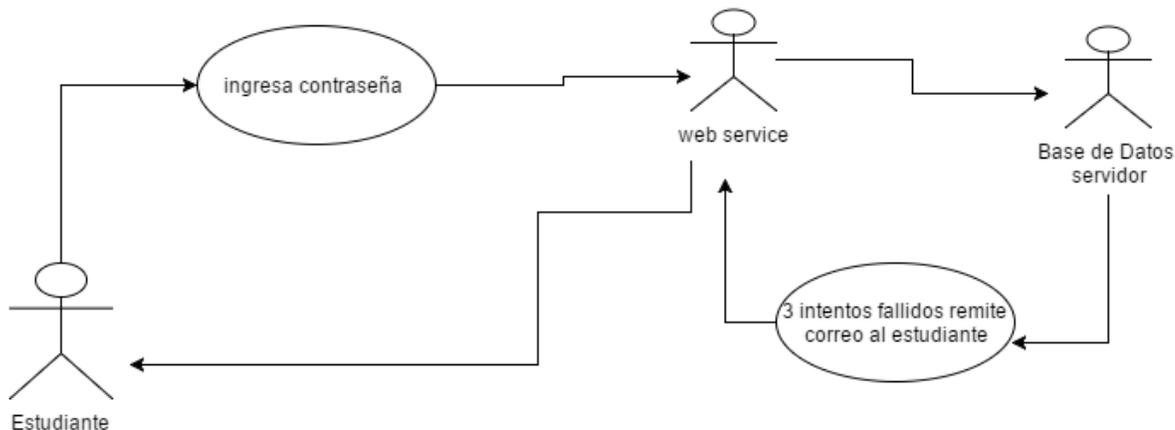


Imagen 17-Contraseña invalidad (CU_REQ_08)

- Versión Android --- IMEI (CU_REQ_09)

| Campo de Caso de uso | Descripción |
|-------------------------------|--|
| Nombre del caso de uso | Versión Android --- IMEI (CU_REQ_09) |
| Actores | Estudiante, WebService, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | Validar versión del dispositivo e IMEI |
| Resumen | Si el estudiante desea descarga la aplicación se comprobara la versión del sistema y su IMEI para un óptimo funcionamiento |
| Precondiciones | el estudiante deberá estar conectado a internet y contar con los requerimientos antes descritos para su funcionamiento |
| Flujo principal | Al momento de la descarga se validara descripción del dispositivo |
| Sub flujo | validar requerimientos del dispositivo |
| Excepciones | |
| PostCondiciones | Antes de la descarga mostrara para que dispositivo está disponible |

Tabla 10-Versión Android --- IMEI (CU_REQ_09)

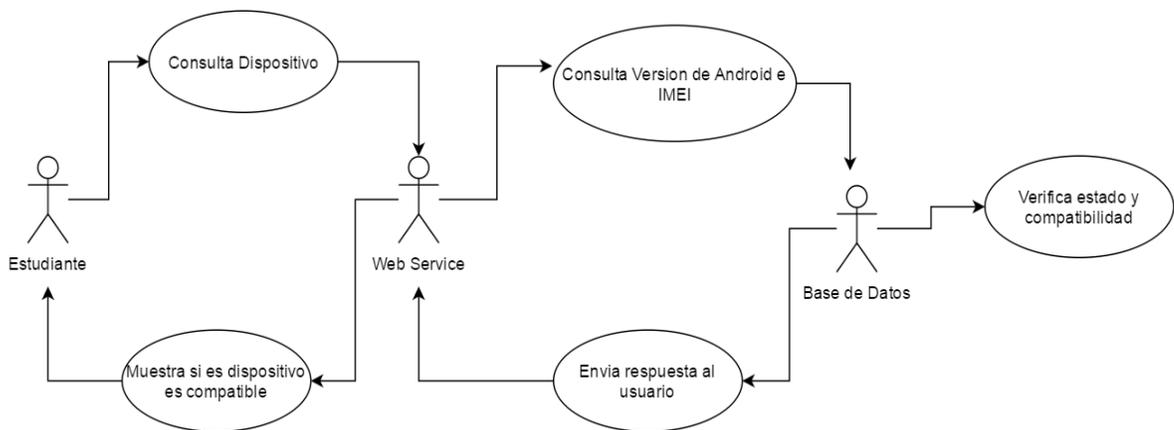


Imagen 18-Versión Android --- IMEI (CU_REQ_09)

- Vinculación, Sincronización y GPS --- IMEI (CU_REQ_10)

| Campo de Caso de uso | Descripción |
|------------------------|---|
| Nombre del caso de uso | Vinculación, Sincronización y GPS --- IMEI (CU_REQ_10) |
| Actores | Estudiante, WebService, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | Hacer vinculación , sincronización del dispositivo y solicitar permisos para acceder al GPS |
| Resumen | Cuando el estudiante ingrese a la aplicación esta comprobará la vinculación de su cuenta y sincronizará esta, también pedirá la activación del GPS para el servicio ingreso a la ucp. |
| Precondiciones | El estudiante deberá tener la aplicación descargada |
| Flujo principal | Al momento de la vinculación la cuenta se sincronizará |
| Sub flujo | Solicitar permisos para activación del GPS |
| Excepciones | NA |
| PostCondiciones | Al usuario le solicitará la activación del GPS para su óptimo funcionamiento(No es obligatorio) |

Tabla 11-Vinculación, Sincronización y GPS --- IMEI (CU_REQ_10)

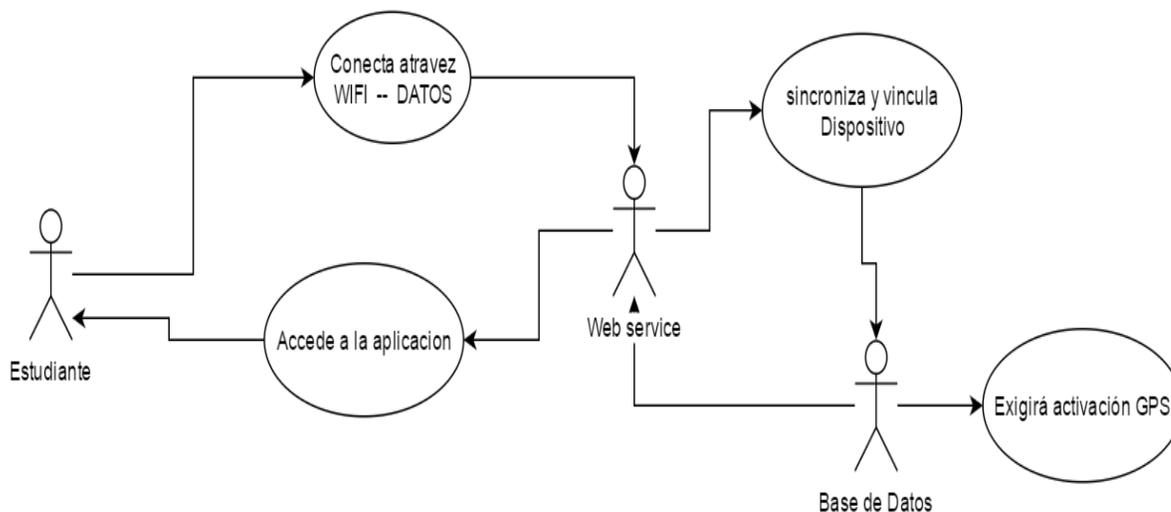


Tabla 12-Vinculación, Sincronización y GPS --- IMEI (CU_REQ_10)

- ConsultarHorarios_Calificaciones_Libros (CU_REQ_11)

| Campo de Caso de uso | Descripción |
|------------------------|--|
| Nombre del caso de uso | Consultar Horarios Calificaciones Libros (CU_REQ_11) |
| Actores | Estudiante, WebService, Base de datos Servidor, Base de datos local |
| Tipo | Primario |
| Propósito | Consultar horarios, calificaciones y libros |
| Resumen | <ul style="list-style-type: none"> • Consultar información de los horarios del estudiante. • Consultar información de las calificaciones del estudiante • Consultar información de los libros de la biblioteca. (SIABUC) |
| Precondiciones | Ninguna |
| Flujo principal | Se presenta al usuario un menú con varias opciones, entre ellas Horarios, calificaciones y SIABUC. Si elige horarios el sistema traerá el horario del estudiante, si se elige calificaciones el sistema consultará las calificaciones del estudiante y si se elige SIABUC el sistema exigirá filtros para la búsqueda del libro. |
| Sub flujo | Validar conexión a internet al buscar libro, verificar que la información este sincronizada |
| Excepciones | Error al buscar libro |
| PostCondiciones | Mensaje de búsqueda exitosa del libro |

Tabla 13 - ConsultarHorarios_Calificaciones_Libros (CU_REQ_11)

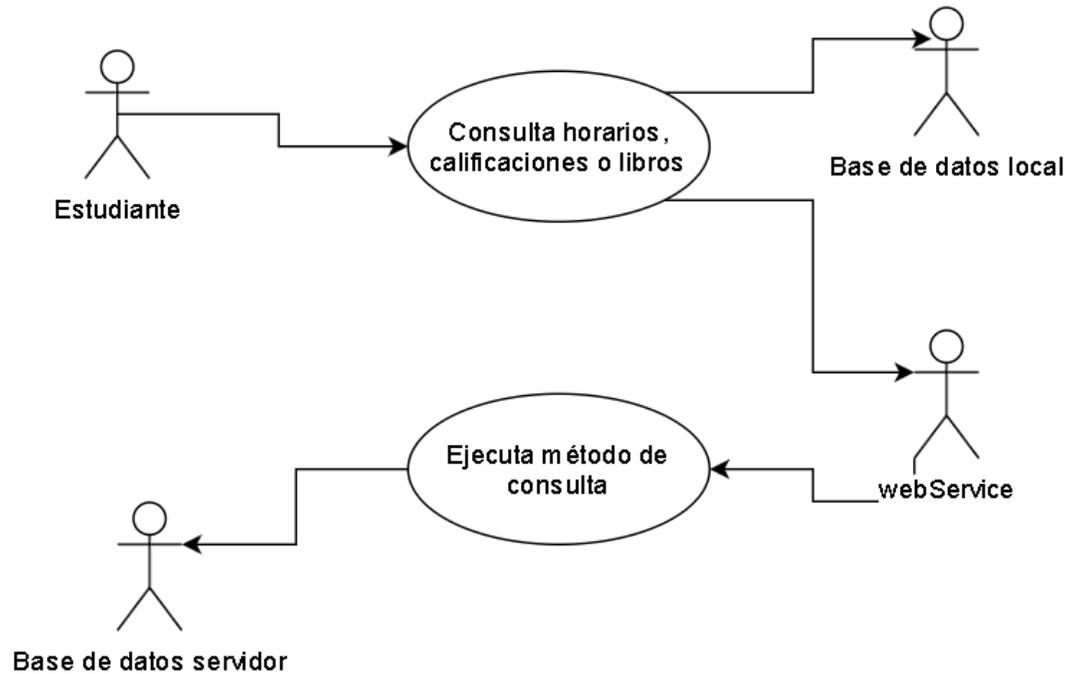


Imagen 19 – Consultar Horarios Calificaciones Libros (CU_REQ_11)

- Noticias (CU_REQ_12)

| Campo de Caso de uso | Descripción |
|------------------------|---|
| Nombre del caso de uso | Noticias (CU_REQ_12) |
| Actores | Estudiante, WebService, Base de datos Servidor, Base de datos local |
| Tipo | Primario |
| Propósito | Publicación y vista de noticias |
| Resumen | Contará con un pequeño formulario móvil y web donde se podrán publicar noticias. Ver noticias de la universidad y la comunidad de la misma. |
| Precondiciones | Ninguna |
| Flujo principal | Se le presenta al usuario un formulario donde se podrán observar las noticias publicadas y una opción donde tendrá la oportunidad de publicar noticias. |
| Sub flujo | Estar loggeado |
| Excepciones | Error al cargar nuevas noticias, Error al publicar noticias |
| PostCondiciones | Mensaje de publicación exitosa |

Tabla 14 - Noticias (CU_REQ_12)

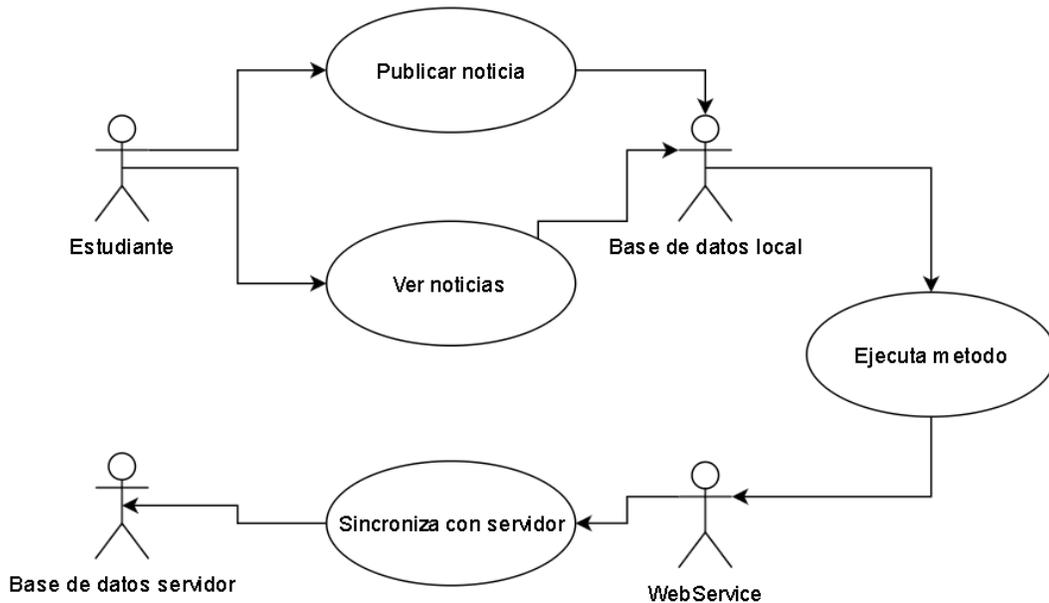


Imagen 20 - Noticias (CU_REQ_12)

- VerHimno_Factultad_Programas (CU_REQ_13)

| Campo de Caso de uso | Descripción |
|------------------------|---|
| Nombre del caso de uso | verHimno_Factultad_Programas (CU_REQ_13) |
| Actores | Estudiante, WebService, Base de datos Servidor, Base de datos local |
| Tipo | Secundario |
| Propósito | Ver información de la universidad |
| Resumen | Ver información de la facultad y de sus respectivos programas. Ver letra del himno de la universidad. |
| Precondiciones | Ninguna |
| Flujo principal | Se le presentara al usuario en el menú 3 opciones, ver facultades, ver programas y ver himnos; Donde el usuario podrá elegir una y le aparecerá información de la opción seleccionada |
| Sub flujo | Estar loggeado |
| Excepciones | Error al cargar información seleccionada |
| PostCondiciones | Muestra información. |

Tabla 15 - verHimno_Factultad_Programas (CU_REQ_13)

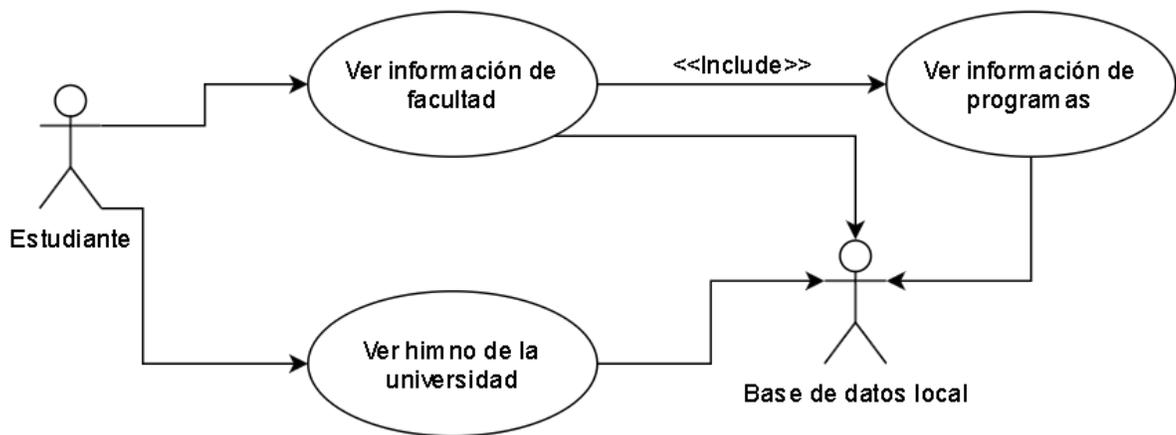


Tabla 16 - verHimno_Facultad_Programas (CU_REQ_13)

- Ayuda aplicativo (CU_REQ_14)

| Campo de Caso de uso | Descripción |
|------------------------|--|
| Nombre del caso de uso | Ayuda (CU_REQ_14) |
| Actores | Estudiante ,aplicativo |
| Tipo | Secundario |
| Propósito | Que el estudiante conozca más acerca de la aplicación |
| Resumen | Si el estudiante desea saber más como fue desarrollada, los términos de condiciones y uso de la app esta la encontrara en el FAQ del dispositivo |
| Precondiciones | Tener la aplicación instalada y haber iniciado sesión |
| Flujo principal | Al momento de saber más del aplicativo este tendrá la opción de mostrar su descripción |
| Sub flujo | Mostrar diverso contenido del aplicativo |
| Excepciones | |
| PostCondiciones | |

Tabla 17 - Ayuda (CU_REQ_14)

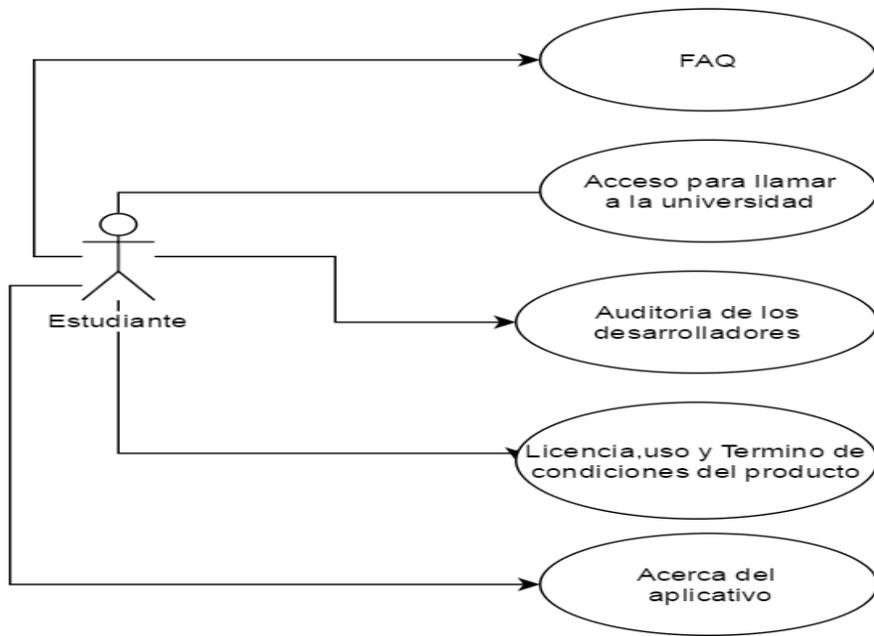


Imagen 21 - Ayuda (CU_REQ_14)

- Diagrama de actividades
- Diagrama de actividades Información UCP y ayuda al usuario

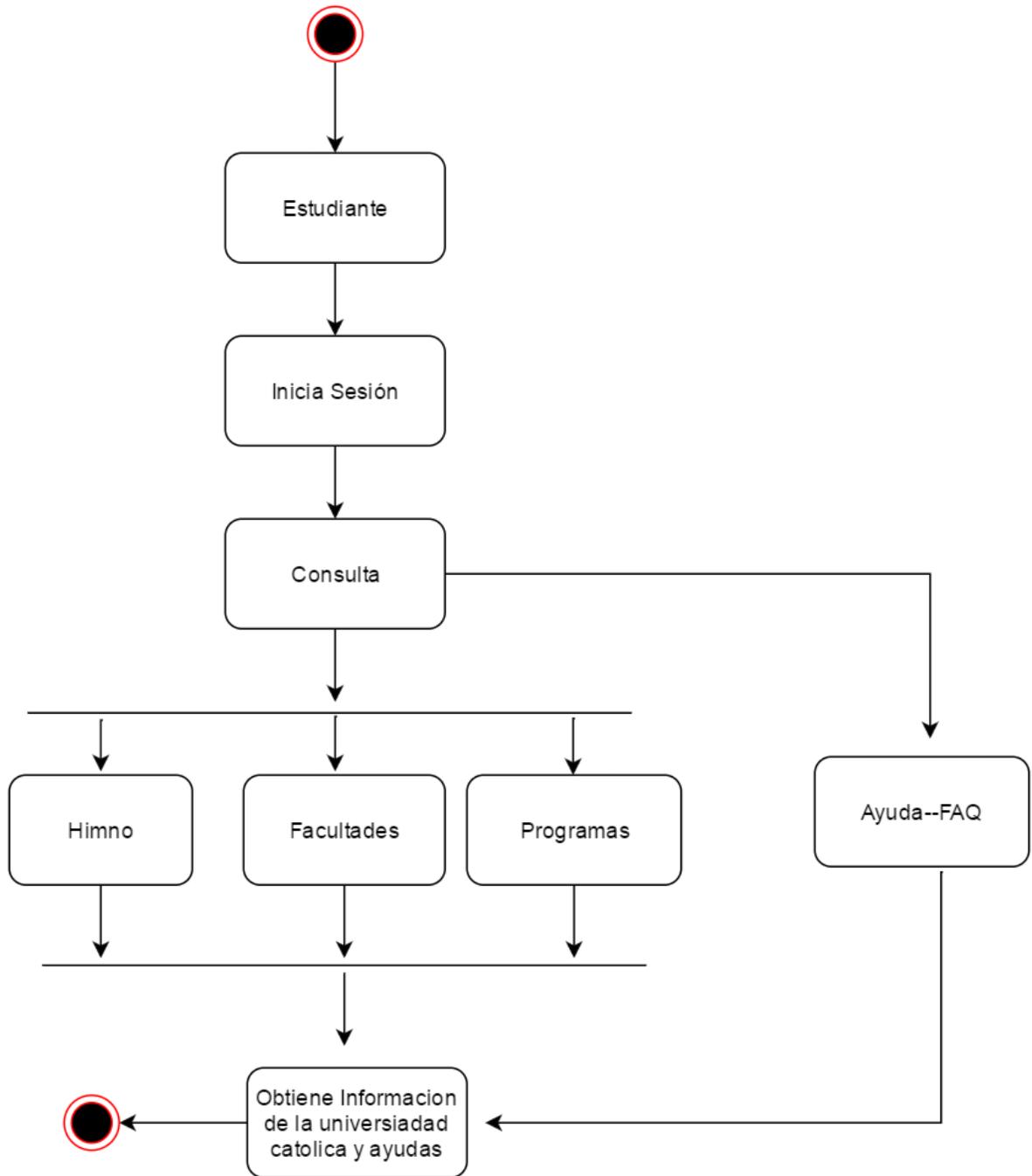


Imagen 22 - Información UCP y ayuda al usuario

| Campo de Diagrama de actividades | Descripción |
|---|---|
| Nombre Diagrama de actividades | Información UCP y ayuda al usuario |
| Actores | Estudiante ,aplicativo |
| Tipo | Secundario |
| Propósito | Que el estudiante conozca más acerca de la aplicación e información de la universidad católica. |
| Resumen | Si el estudiante desea saber más como fue desarrollada, los términos de condiciones y uso de la app esta la encontrara en el FAQ del dispositivo adicionalmente encontrara todo lo relacionado con la universidad católica de Pereira |
| Precondiciones | tener la aplicación instalada y haber iniciado sesión |
| Flujo principal | Al momento de saber más del aplicativo este tendrá la opción de mostrar su descripción |
| Sub flujo | Mostrar diverso contenido del aplicativo |
| Excepciones | |
| PostCondiciones | |

Tabla 18 - Información UCP y ayuda al usuario

- Diagrama de actividades Sincronización GPS, IMEI y consultas

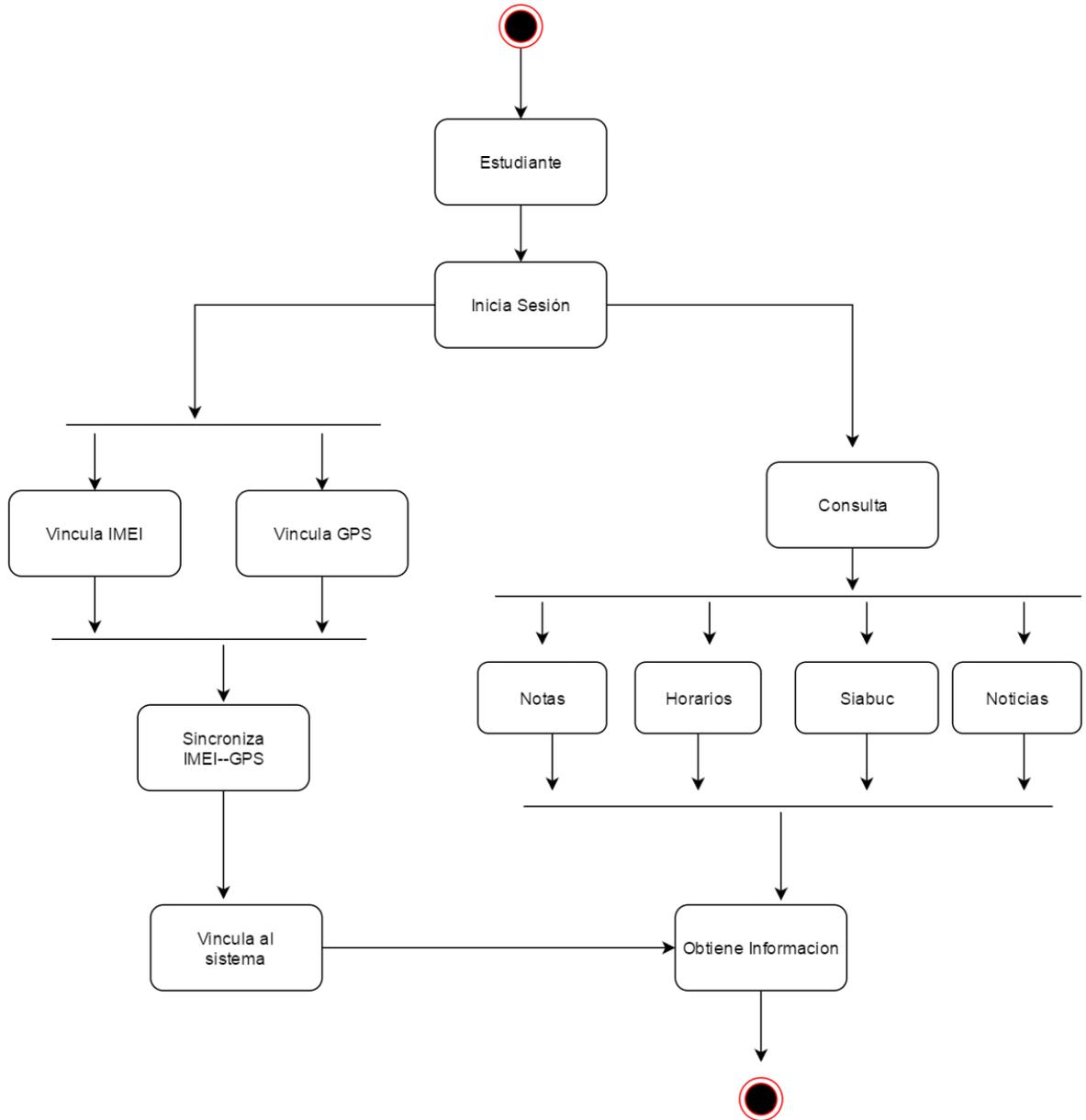


Imagen 23 - Sincronización GPS, IMEI y consultas

| Campo de Diagrama de actividades | Descripción |
|--|--|
| Nombre Diagrama de actividades de | Sincronización GPS, IMEI y consultas |
| Actores | Estudiante, WebService, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | Sincronizar equipo y GPS adicional realizar consultas |
| Resumen | Al usuario iniciar sesión el sistema sincronizara el equipo y activara el GPS una vez esto el |
| Precondiciones | El estudiante deberá tener su cuenta y contraseña asociada |
| Flujo principal | Al momento que el usuario ingresa a la aplicación deberá registrarse con su usuario y contraseña |
| Sub flujo | Sincronizar el equipo para que al usuario pueda realizar las consultas |
| Excepciones | N/A |
| PostCondiciones | El estudiante previamente deberá contar con sesión inicializada |

Tabla 19 - Sincronización GPS, IMEI y consultas

- Diagrama de actividades Información personal opciones de idioma y vinculación de estudiante

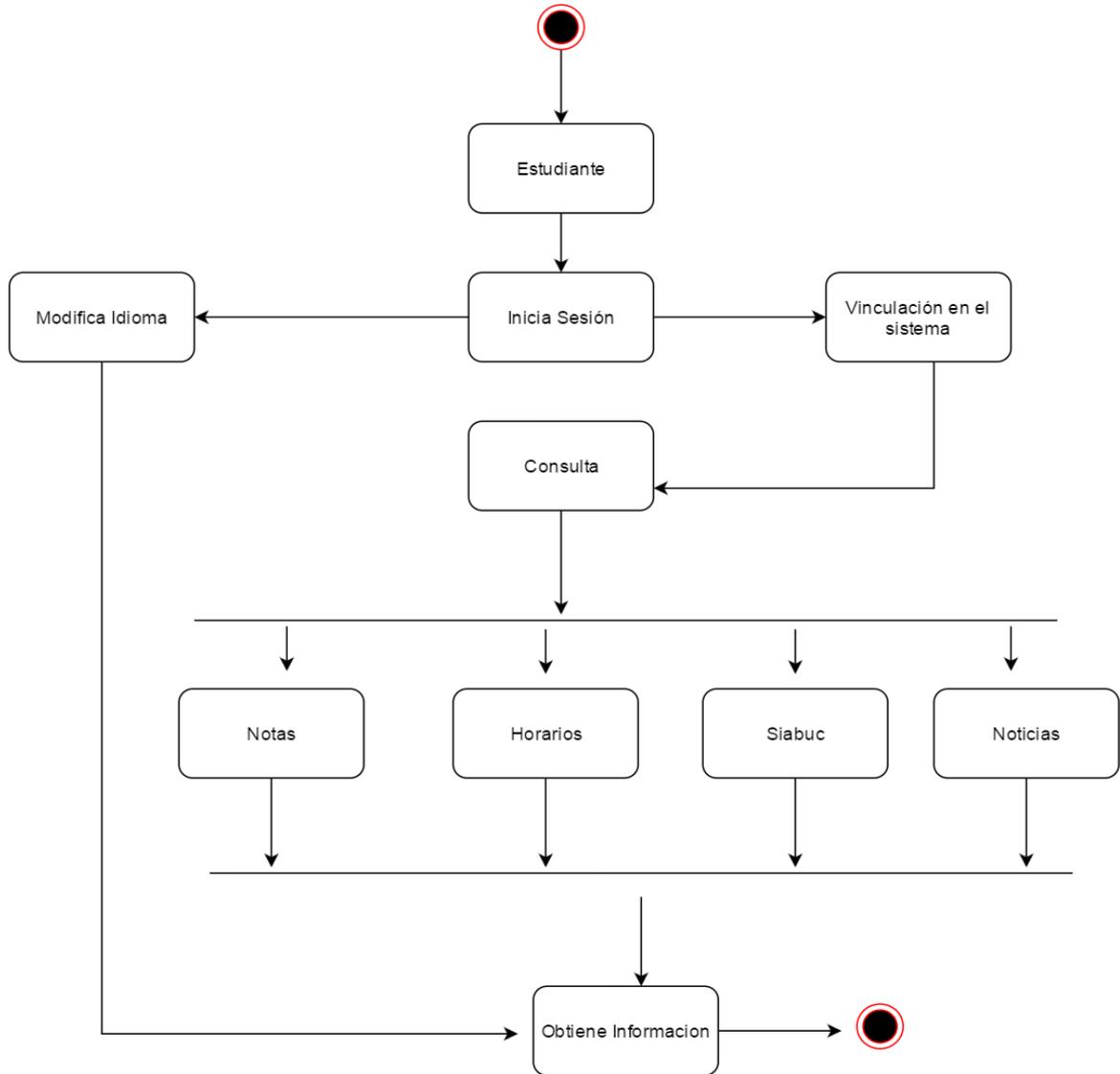


Imagen 24 - Información personal opciones vinculación de estudiante

| Campo de Diagrama de actividades | Descripción |
|---|--|
| Nombre Diagrama de actividades | Diagrama de actividades Información personal opciones de idioma y vinculación de estudiante |
| Actores | Estudiante, WebService, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | El estudiante pueda hacer modificaciones de idioma y saber su información personal de la universidad |
| Resumen | El estudiante pueda hacer modificaciones de idioma a su gusto y podrá realizar consultas respecto a la información personal asociada a su sesión |
| Precondiciones | El estudiante deberá tener su cuenta y contraseña asociada |
| Flujo principal | al momento que el usuario ingresa a la aplicación deberá registrarse con su usuario y contraseña |
| Sub flujo | Sincronizar el equipo para que al usuario pueda realizar las consultas |
| Excepciones | N/A |
| PostCondiciones | El estudiante previamente deberá contar con sesión inicializada |

Tabla 20 - Información personal opciones vinculación de estudiante

- Diagrama de actividades Sincronización y verificación de usuario y contraseña

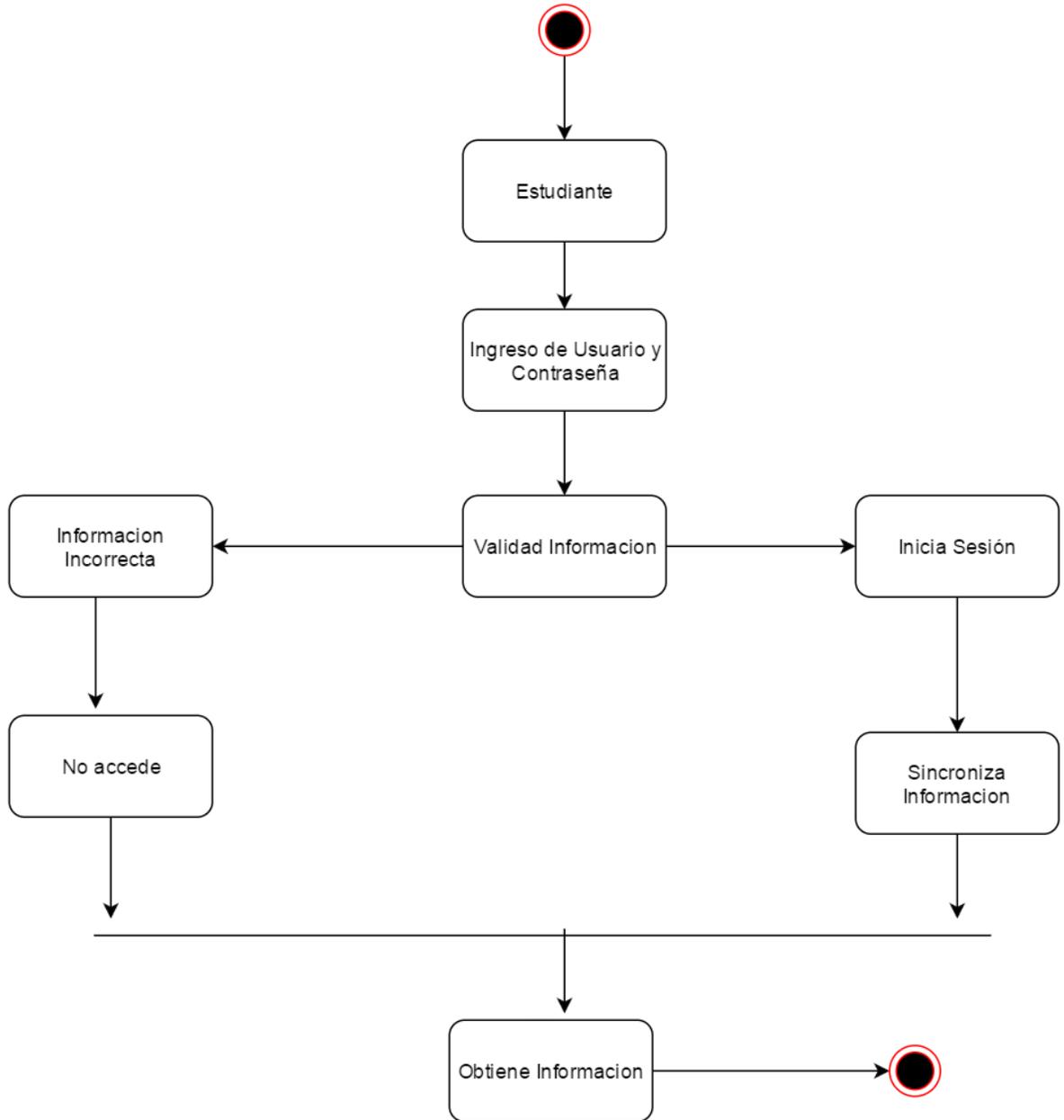


Imagen 25 - Sincronización y verificación de usuario y contraseña

| Campo de Diagrama de actividades | Descripción |
|---|---|
| Nombre del caso de uso | Sincronización y verificación de usuario y contraseña |
| Actores | Estudiante, WebService, Base de datos local, Base de datos Servidor. |
| Tipo | Secundario |
| Propósito | Agregar o cambiar Contraseña y sincronización de la información |
| Resumen | Agregar o cambiar o Contraseña del usuario y sincronización de la información relacionada a la cuenta del estudiante |
| Precondiciones | El estudiante tenga su APP instalada y tener un usuario y contraseña validos |
| Flujo principal | Se presenta al usuario una pantalla con 2 opciones: "PIN" y "Contraseña"; Si se elige la opción "PIN", se abra una ventana para que el usuario agregue o cambie su PIN. Si se elige la opción "Contraseña", Aparecerá una ventana para que el usuario cambie su contraseña. |
| Sub flujo | Estar registrado en las bases de datos como estudiante activo; Tener cuenta vinculada; Tener sesión iniciada. |
| Excepciones | Error al verificar PIN |
| PostCondiciones | Mostrar mensaje de error al cambiar PIN o Contraseña; Mostrar mensaje de error al verificar que el PIN o Contraseña Coincidan; Mostrar mensaje de cambio exitoso. |

Tabla 21 - Sincronización y verificación de usuario y contraseña

2.4.3. Nomenclatura

- Componentes de diseño

| TIPO DE COMPONENTE | NOMBRE DE COMPONENTE | |
|---------------------------|-----------------------------|---------------------|
| Buttons | Btn | + nombre del objeto |
| Labels | Lb | + nombre del objeto |
| Textview | Txv | + nombre del objeto |
| EditText | Et | + nombre del objeto |
| Fragmettabhost | Fth | + nombre del objeto |
| Tab widget | tw | + nombre del objeto |
| Radio button | rbtn | + nombre del objeto |
| Imageview | imgv | + nombre del objeto |
| Checkbox | chk | + nombre del objeto |
| Imagbutton | imgb | + nombre del objeto |
| Webview | wv | + nombre del objeto |
| Spinner | spn | + nombre del objeto |
| Activities | activity_ | + nombre del objeto |

Tabla 22 - Componentes de diseño

- Variables

| TIPO DE DATO | NOMBRE DE VARIABLE | |
|---------------------|---------------------------|---------------------|
| Integer | int | + nombre del objeto |
| Long | lg | + nombre del objeto |
| Double | dou | + nombre del objeto |
| Float | flt | + nombre del objeto |
| Short | sht | + nombre del objeto |
| Byte | bte | + nombre del objeto |
| Boolean | bool | + nombre del objeto |
| Char | ch | + nombre del objeto |
| String | str | + nombre del objeto |
| Array | vector---matrix | + nombre del objeto |

Tabla 23 – Variables

- Base de datos

| TIPO DE OBJETOS | NOMBRE DE OBJETO | |
|----------------------------|------------------|------------------------|
| Tablas | T | + nombre del objeto_PA |
| Campos | Iniciales tabla | + nombre del objeto_PA |
| Procedimientos almacenados | Iniciales tabla | + nombre del objeto_PA |
| Parámetros | Iniciales tabla | + nombre del objeto_PA |
| Funciones | Fnc | + nombre del objeto_PA |
| Vistas | Vst | + nombre del objeto_PA |
| Variables de tabla | vtable | + nombre del objeto_PA |
| Tablas temporales | ttemp | + nombre del objeto_PA |

Tabla 24 - Base de datos

2.4.4. Esquemas de datos

- Esquema conceptual



Imagen 26 - Esquema Conceptual

- Modelo Entidad Relación

En el desarrollo del software se crearon dos bases de datos, Una para el cliente y otra para el servidor.

- Entidad Relación Cliente

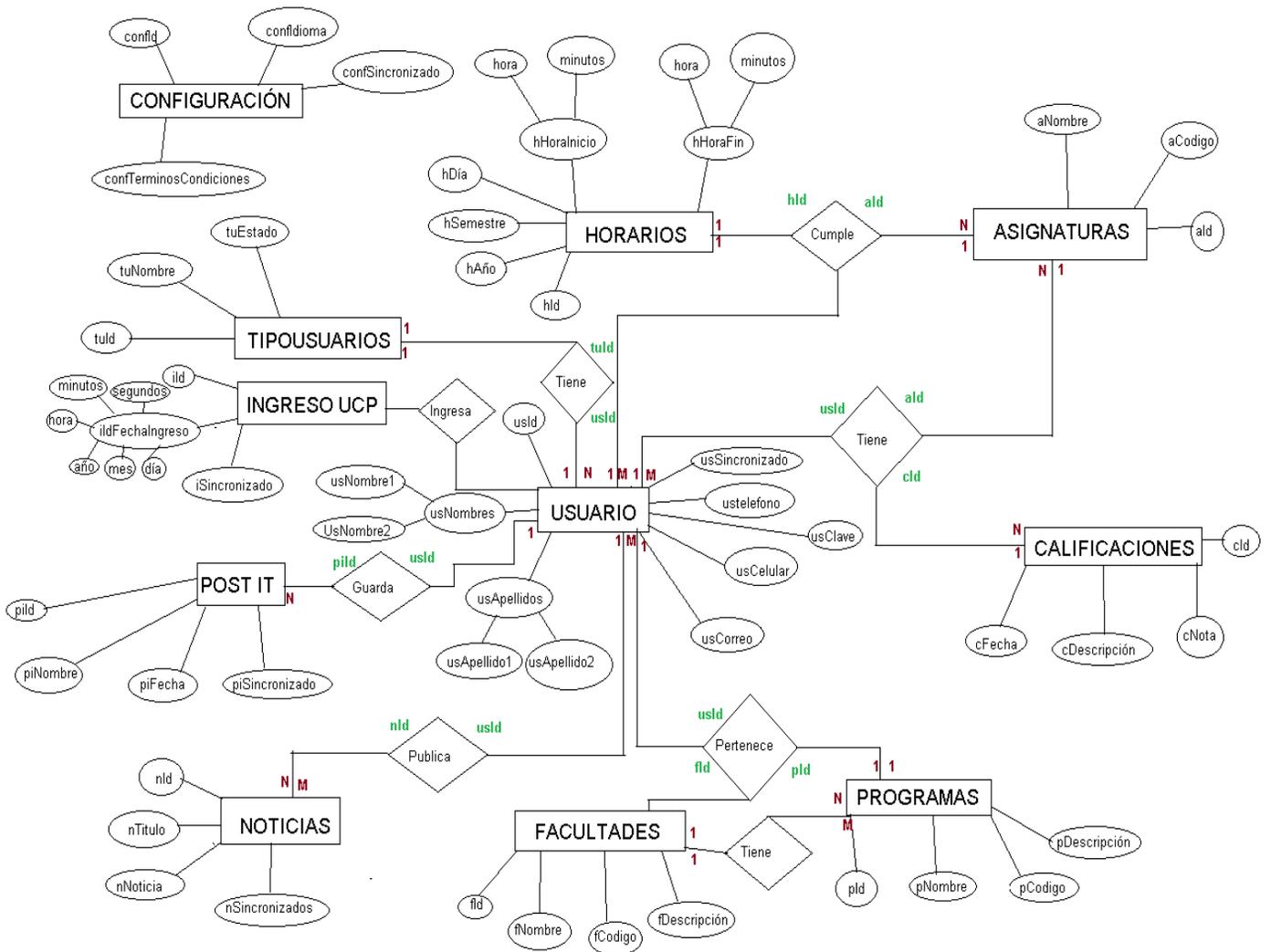


Imagen 27 - Entidad Relación Cliente

- Entidad Relación Servidor (Prototipo)

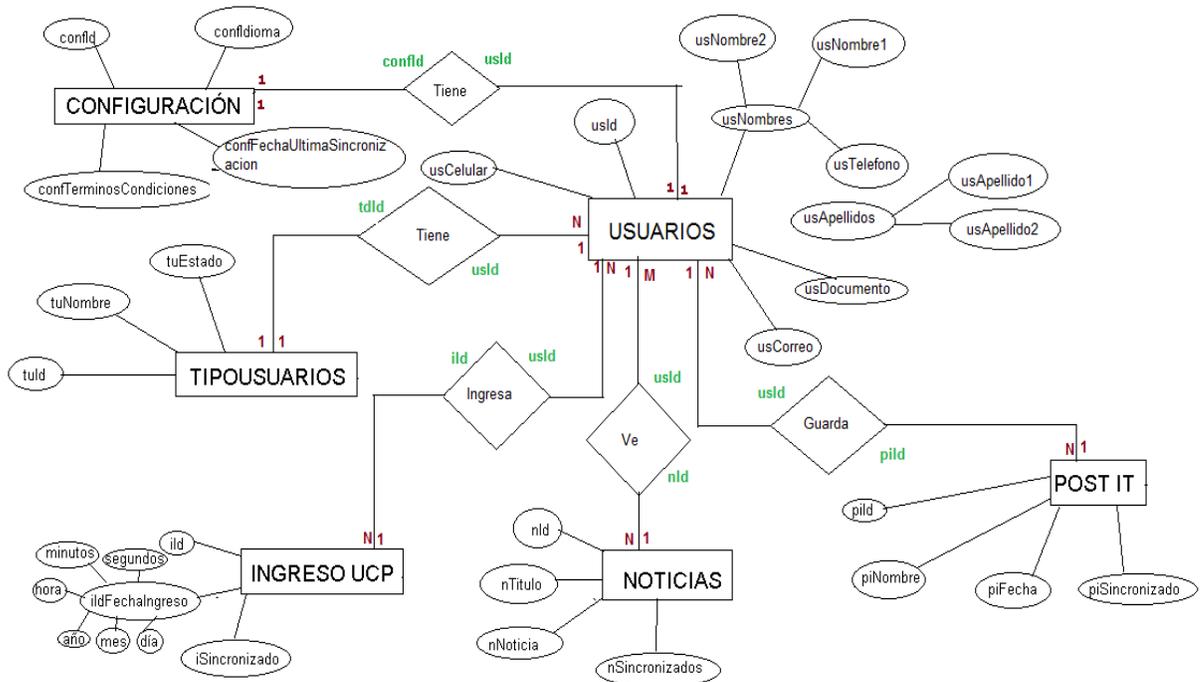


Imagen 28 - Entidad Relación Servidor (Prototipo)

- Modelo Relacional
 - Modelo Relacional Servidor

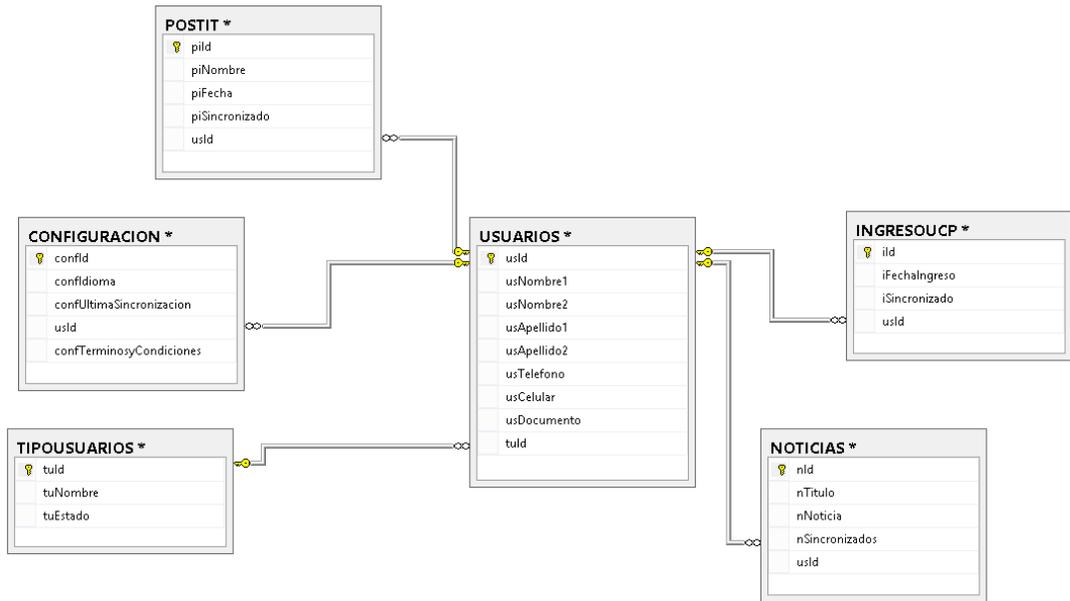


Imagen 29 - Modelo Relacional Servidor (Prototipo)

2.4.5. Interfaz de usuario

Las siguientes imágenes muestran algunas vistas de la interfaz que verá el usuario al momento de utilizar el aplicativo móvil UCP-PIM.



Imagen 30 - Interfaz - Registrar



Imagen 31 - Interfaz - Ingresar

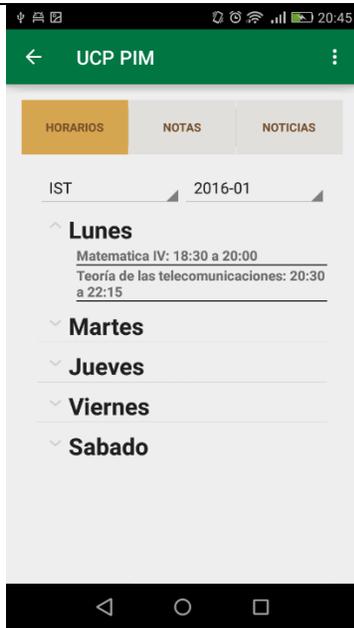


Imagen 32 - Interfaz - Horarios



Imagen 33 - Interfaz - Notas

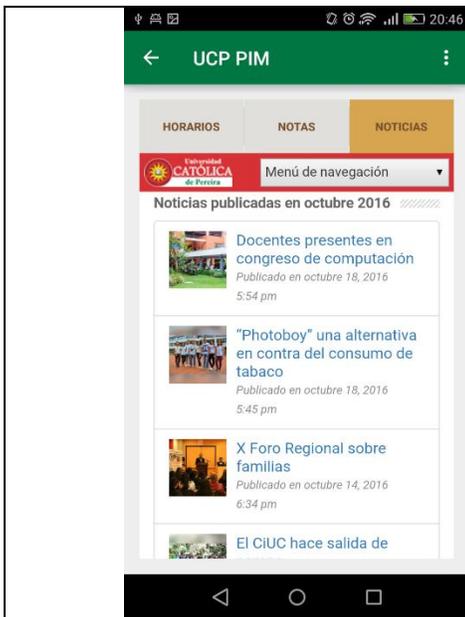


Imagen 34 - Interfaz - Noticias

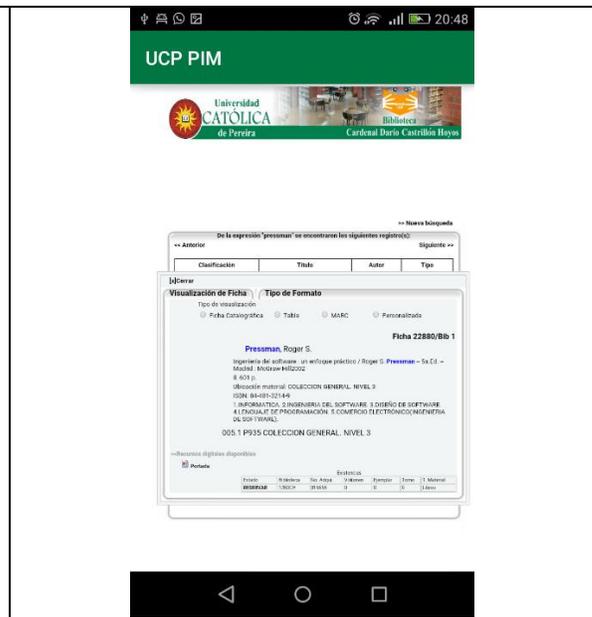


Imagen 35 - Interfaz - SIABUC



Imagen 36 - Interfaz - Mi Universidad

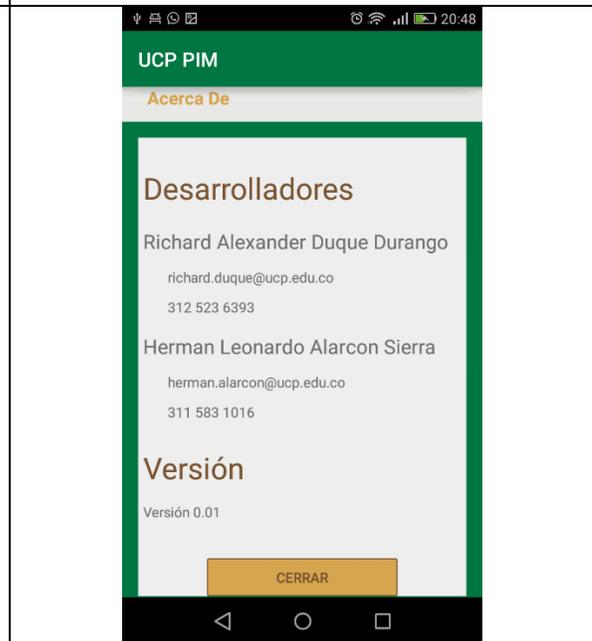


Imagen 37 - Interfaz - Acerca de

Tabla 25 - Interfaz de usuario

2.4.6. Plantilla de pruebas

| | | | | | |
|----------------------------|---|-----------------|--------------|------------------|--------------------------------|
| Proyecto: | UCP-PIM | | | | |
| Versión: | 1.0 | | | | |
| Creador | Herman Alarcón - Richard Duque | | | | |
| Descripción | Documento para pruebas unitarias del proyecto UCP-PIM | | | | |
| Funcionalidad | Caso de prueba | Registro | Fecha | ¿Aprueba? | Prueba de funcionalidad |
| Nombre de la funcionalidad | Caso de prueba 1 para la funcionalidad | 1 | dd/mm/yyyy | Si/No | Ver registro |
| | Caso de prueba 2 para la funcionalidad | 2 | dd/mm/yyyy | Si/No | Ver registro |
| Funcionalidad | Caso de prueba | * | Fecha | ¿Aprueba? | Prueba de funcionalidad |
| Nombre de la funcionalidad | Caso de prueba 1 para la funcionalidad | 3 | dd/mm/yyyy | Si/No | Ver registro |
| | Caso de prueba 2 para la funcionalidad | 4 | dd/mm/yyyy | Si/No | Ver registro |
| | Caso de prueba 3 para la funcionalidad | 5 | dd/mm/yyyy | Si/No | Ver registro |

3.0. CAPITULO 3

3.1. METODOLOGÍA

Basados en la problemática de la Universidad Católica de Pereira y la solución móvil planteada, es seleccionada la metodología Mobile D por permitir un desarrollo fácil, rápido e incremental y una programación en parejas reduciendo así el desarrollo a prueba-error.

Ciclo de desarrollo de Mobile-D



Imagen 38 - Mobile-D

Exploración:

En este punto se analiza la necesidad, se identifican los interesados en el proyecto y se define un alcance.

- Por medio de observación se identifica lo complejo que es para el estudiante de la universidad católica de Pereira estar al tanto de información como Notas, Horarios, Noticias y al momento de consultar libros.
- Se define el alcance teniendo en cuenta las restricciones que se podían tener al momento del desarrollo.
- Se realiza un análisis sobre los posibles StakeHolder que pueden intervenir en el pre Desarrollo, post desarrollo y producción.

Inicializar:

En esta etapa se identifican todos los recursos para el proyecto, incluyendo al talento humano y se planifican las tareas a realizar en fases posteriores

- Identificar los recursos que se tienen para el desarrollo.
- Planificar las tareas a realizar
- Realizar asignaciones de las tareas a los integrantes del equipo de proyecto teniendo en cuenta las actitudes y aptitudes de cada uno.

Producción:

En esta etapa se inicia el desarrollo del aplicativo y a su vez se documentan las funcionalidades en el transcurso de este, obteniendo en esta etapa la producción de cada una de las funcionabilidades para el manejo y desempeño del aplicativo para que este pueda funcionar óptimamente.

- Identificar las funciones del aplicativo a desarrollar.
- Desarrollar interfaces, métodos y funciones para cada proceso.
- Mostrar los resultados obtenidos en esta en el desarrollo del aplicativo en cada uno de sus procesos requeridos

Estabilidad

En esta etapa se crea la integración de cada función y proceso al resto del aplicativo, esta a su vez se encarga de que cada función integrada se ejecute correctamente con el resto de la aplicación obteniendo su estabilidad con las demás funciones ya integradas de este, también esta etapa se desarrolla en cada función agrada.

- Implementar cada función con la aplicación.
- Verificar su estabilidad con las demás funciones ya integradas.
- Realizar en cada función su funcionabilidad y adecuación con los demás procesos.

Test y reparación del sistema

En esta etapa se realizara pruebas al aplicativo para observar su estabilidad, funcionabilidad en cada proceso asignado, verificando su comportamiento y validando si este no posee errores inesperados con el fin de solucionarlos y así

poder entregar un producto estable, completó y funcional en cada una de las funciones asignadas.

- Realización de pruebas en el aplicativo.
- Verificación de errores y corrección de ellos.
- Análisis de cada función asignada.
- Análisis de la estabilidad, funcionalidad del aplicativo completamente para su uso con el usuario final.

3.1.1. Se aplicó ingeniería del software al proyecto.

- Obtener los requisitos, con el tema principal que motiva el inicio del estudio y creación del nuevo software o modificación de uno ya existente.
- Identificar los recursos que se tienen, en esto entra el conocer los recursos humanos y materiales que participan en el desarrollo de las actividades.
- Analizar los requisitos obtenidos para iniciar con la creación con estas se identifica las necesidades del usuario y poder satisfacer estas.
- Utilizar casos de usos y diagramas UML para facilitar el desarrollo con los cuales podemos obtener una mejor aplicabilidad de ingeniería en el proyecto.
- Crear diccionario de datos y Diseñar base de datos para también facilitar el desarrollo y tener una documentación más avanzada en cada uno de los procesos y funciones del aplicativo.

3.1.2. Realizar un estudio sobre arquitectura de software e implementar dicha arquitectura para tener una mejor estructuración del proyecto.

- Buscar en internet, libros y preguntando a otros programadores como montar una buena arquitectura de software.
- Investigar cuándo y por qué se deben usar 3 o más capas en la arquitectura del proyecto
- Implementar arquitectura de software para iniciar un buen desarrollo ordenado.

3.1.3. Diseñar prototipo de software para un fácil manejo basándonos en encuestas.

- Realizar encuestas por medio de formulario de Google para identificar para el usuario que sistemas de los mencionados en la encuesta son de fácil manejo e identificar cuál de estos aplicar en UCP-PIM.

| | |
|------------------------------------|------------------------------------|
| Muestra | 51 personas |
| Población objeto de estudio | Universidad Católica de Pereira |
| Forma de contacto | Formulario de Google |
| Selección de la muestra | Muestreo Probabilístico Aleatorio. |
| Rango de fecha de encuesta | 2016-09-22 a 2016-10-30 |

Tabla 26 - Ficha técnica encuesta diseño fácil manejo.

¿Cree usted que el aplicativo móvil WhatsApp es de fácil manejo? (51 respuestas)

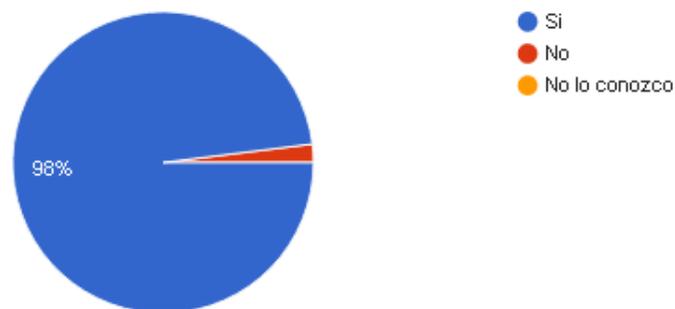


Imagen 39 - Pregunta 1 Encuesta

¿Cree usted que el aplicativo FaceBook es de fácil manejo? (51 respuestas)

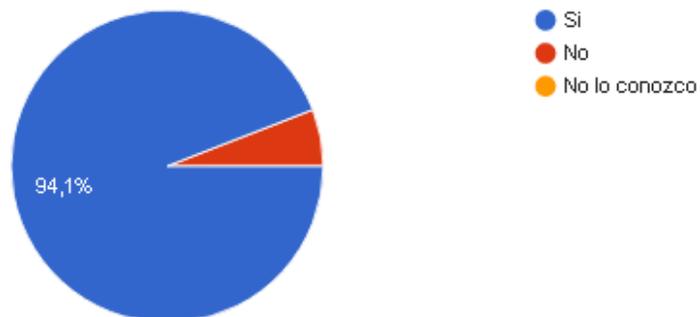


Imagen 40 - Pregunta 2 Encuesta

¿Cree usted que el aplicativo twitter es de fácil manejo? (51 respuestas)

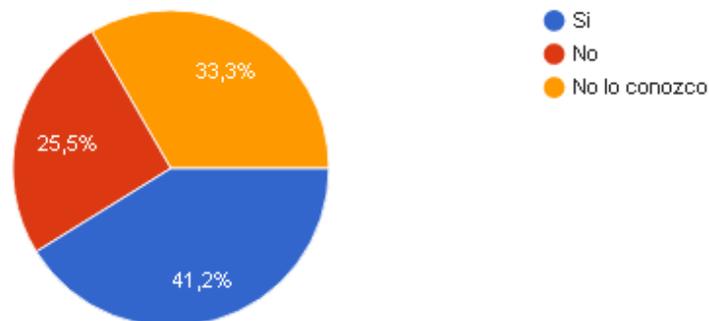


Imagen 41 - Pregunta 3 Encuesta

¿Cree usted que el aplicativo Instagram es de fácil manejo? (51 respuestas)

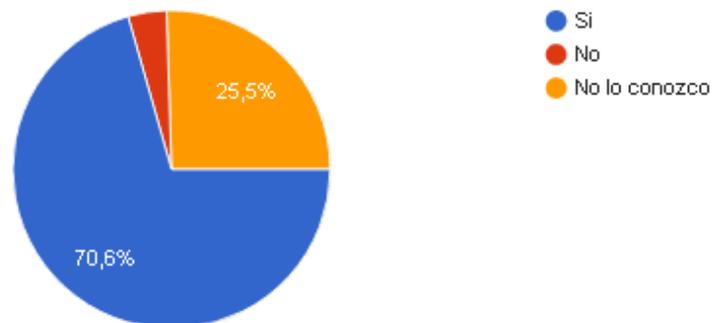


Imagen 42 - Pregunta 4 Encuesta

¿De los aplicativos mencionados anteriormente cual es de mas fácil manejo y conoce la mayor parte de sus funciones?

(51 respuestas)

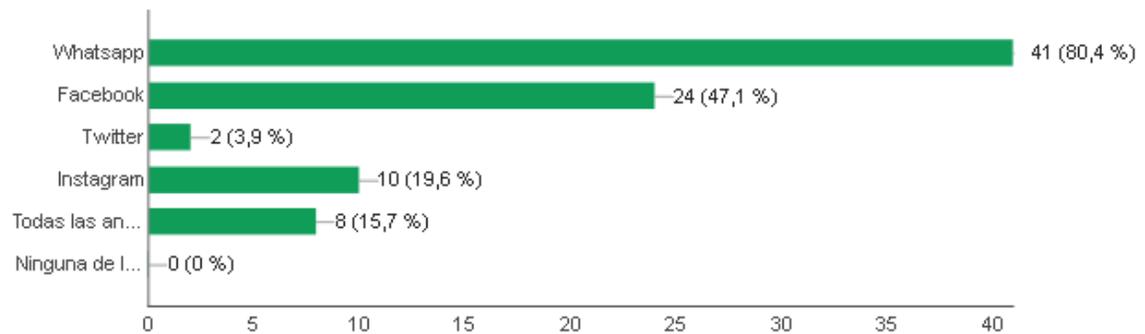


Imagen 43 - Pregunta 5 Encuesta

¿Qué revisa usted mas cuando ingresa a la página de la Universidad Católica de Pereira?

(51 respuestas)

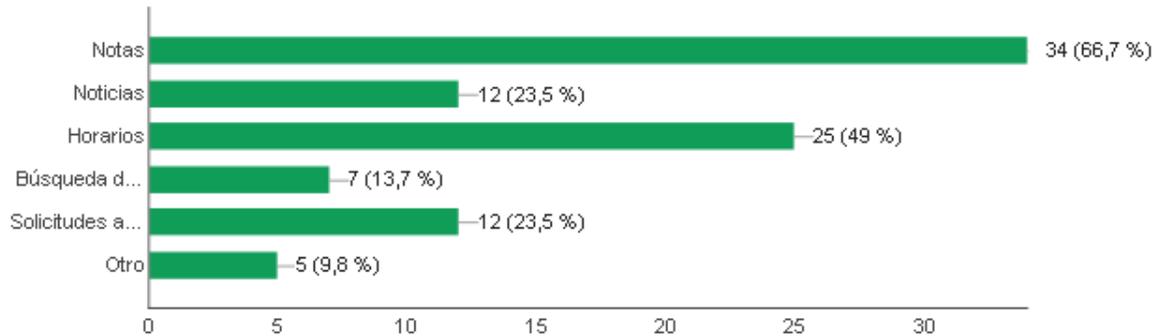


Imagen 44 - Pregunta 6 Encuesta

¿Cree que el siguiente prototipo de diseño de aplicativo móvil para la universidad Católica de Pereira sería de fácil manejo?

(51 respuestas)

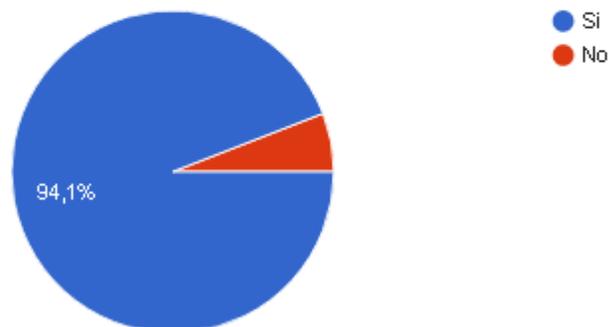


Imagen 45 - Pregunta 7 Encuesta

- Buscar información sobre cómo posicionar los componentes del diseño en cada uno de los formularios de acuerdo a una media.
- Estudiar e implementar funcionalidades atractivas para los usuarios, por ejemplo accesos directos
- Antes de empezar con el diseño del software se buscara que este se ha de fácil uso para el usuario, ya que cualquier funcionamiento complejo puede causar la inutilidad del software.

- Iniciar diseño del aplicativo buscando funcionalidades atractivas y agradables para el uso de este.

3.1.4. Codificar de acuerdo a lo encontrado en el objetivo 1.

- Definir arquitectura del software para su fácil desarrollo.
- Implementar nomenclatura que se va a utilizar en la declaración de variables, objetos, campos, tablas, base de datos para llevar un orden durante el desarrollo.
- Desarrollo del aplicativo móvil UCP-PIM.

3.1.5. Realizar pruebas al software para garantizar su funcionamiento.

- Investigar sobre un buen método para realizar pruebas a un software, teniendo en cuenta la calidad y estabilidad.
- Realizar pruebas, búsqueda de errores y corrección de los mismos.

3.1.6. Implantar versión beta en Google Play Store

- Comprar cuenta de desarrollador de Google Play Store.
- Consultar sobre como publicar un aplicativo móvil en Google
- Publicar versión Beta del aplicativo

3.2. PLANIFICACIÓN DEL PROYECTO

3.2.1. GANTT

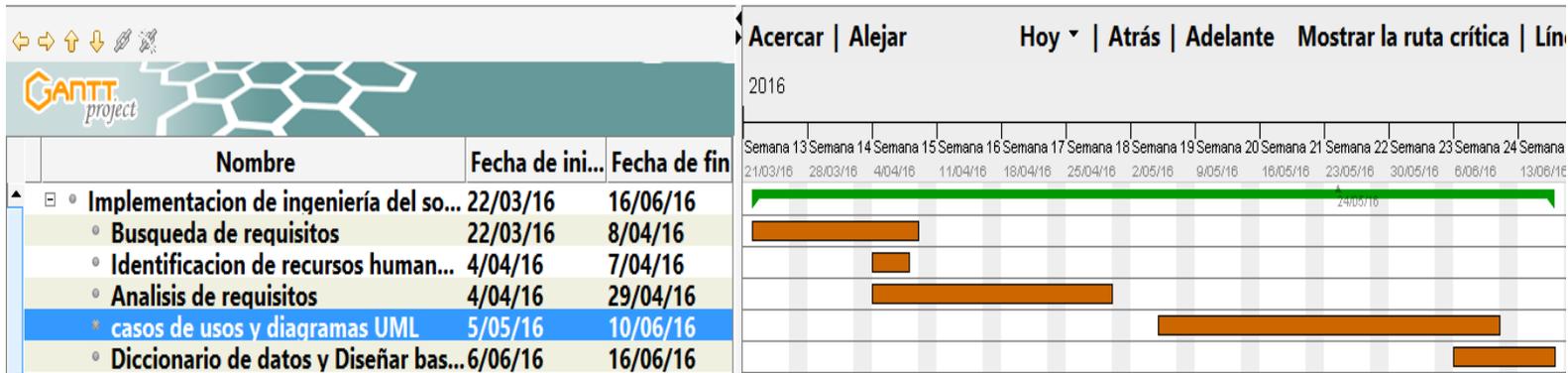


Imagen 46 – Cronograma GANTT 01

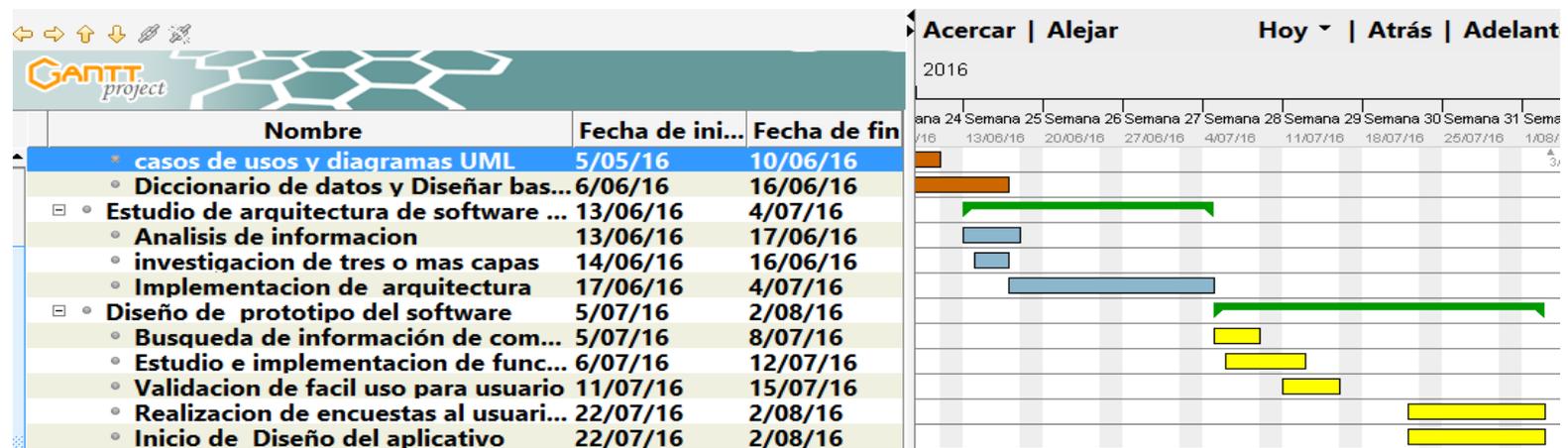


Imagen 47 – Cronograma GANTT 02

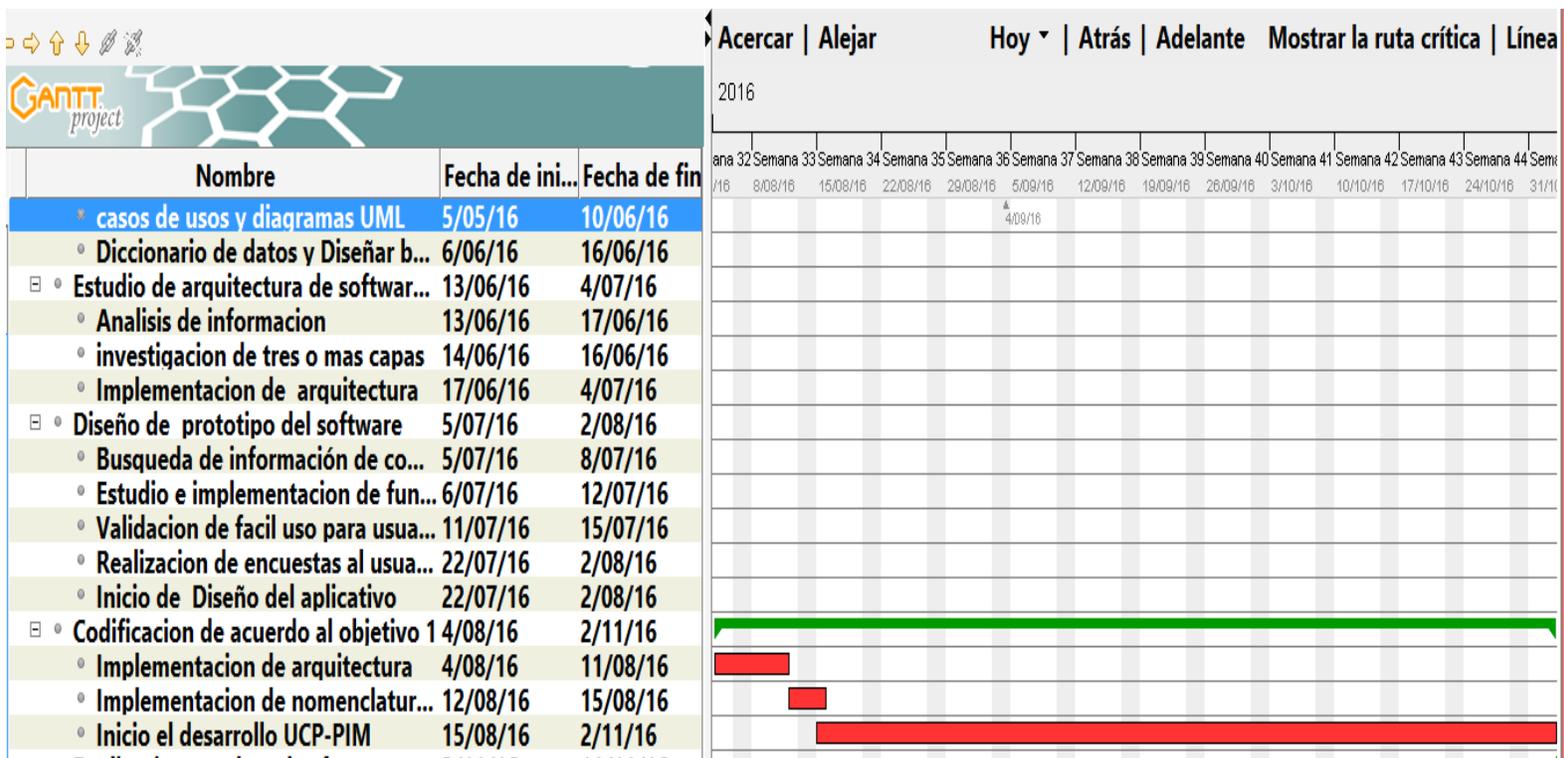


Imagen 48 – Cronograma GANTT 03

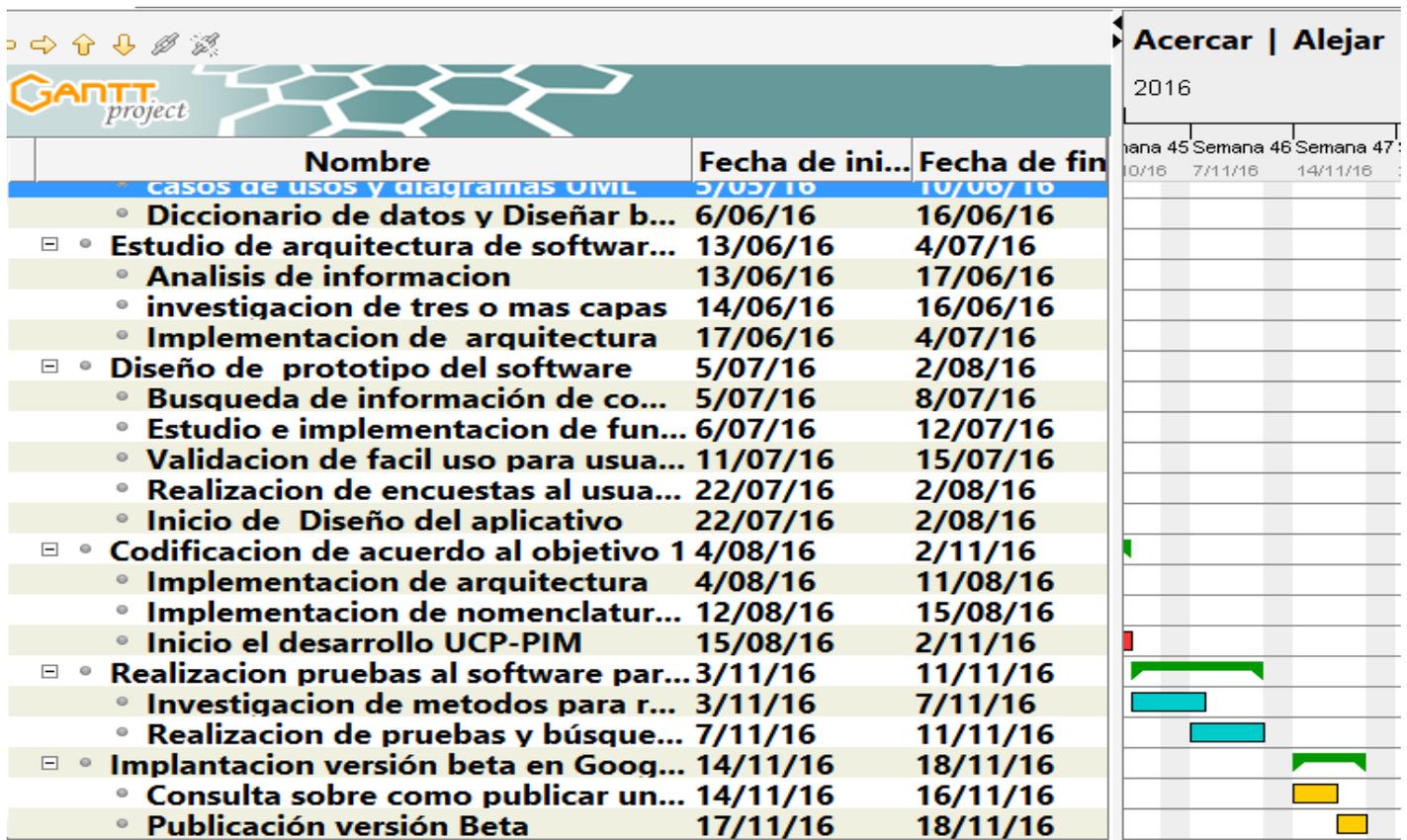


Imagen 49 – Cronograma GANTT 04

3.2.2. PERT

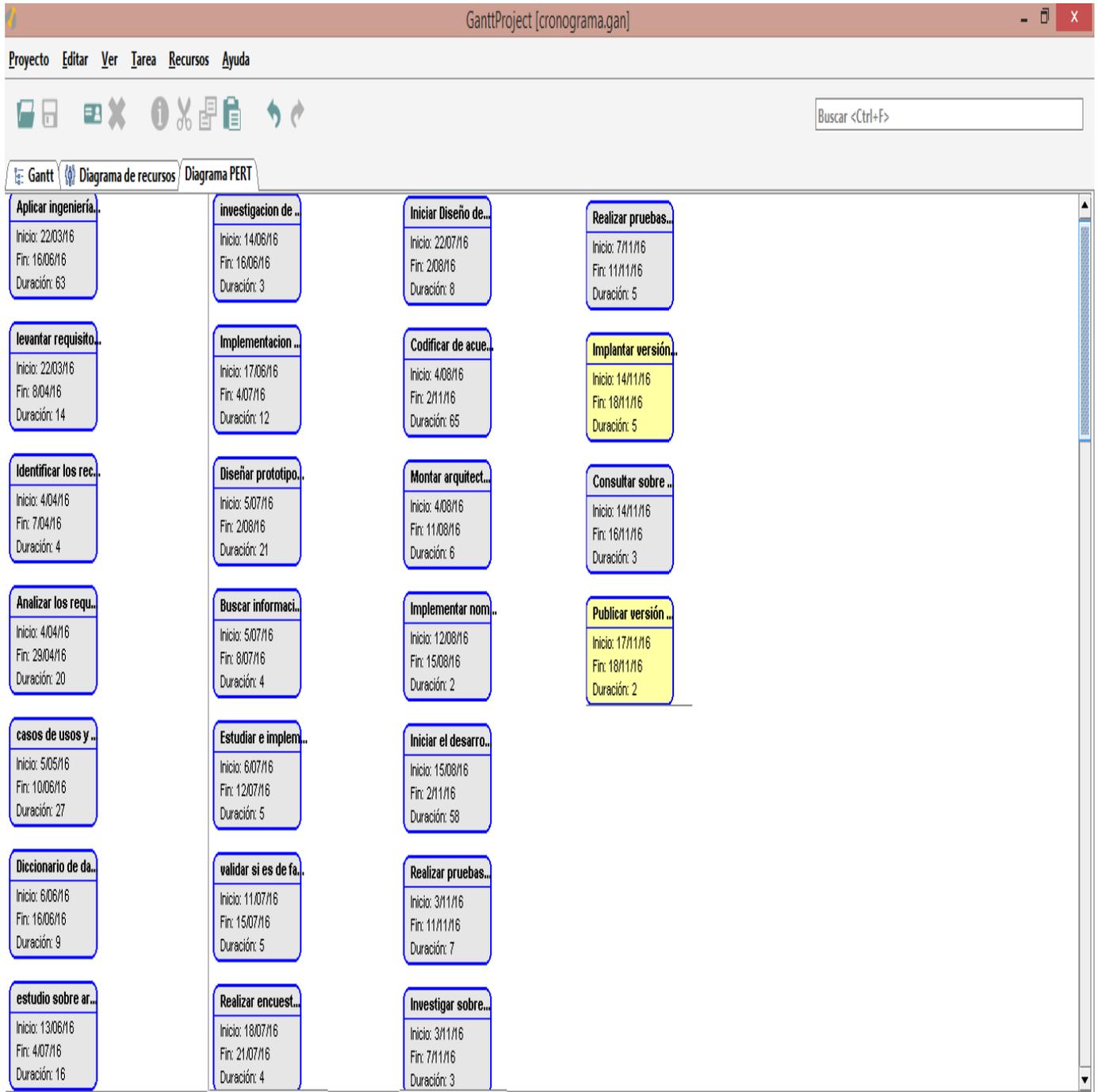


Imagen 50 - Cronograma PERT

3.3. CRONOGRAMA

Para un total de 210 días, a continuación el cronograma de actividades

| Aplicar ingeniería del software a este proyecto. | | | | |
|--|------------------------------|--------------|------------|------|
| Actividad | Responsable | Fecha inicio | Fecha Fin | Días |
| Obtener los requisitos, es decir, el tema principal que motiva el inicio del estudio y creación del nuevo software o modificación de uno ya existente. | Richard Duque-Herman Alarcón | 2016-03-22 | 2016-04-05 | 14 |
| Identificar los recursos que se tienen, en esto entra el conocer los recursos humanos y materiales que participan en el desarrollo de las actividades. | Richard Duque-Herman Alarcón | 2016-04-06 | 2016-04-10 | 4 |
| Analizar los requisitos obtenidos para iniciar con la creación; Identificar las necesidades del usuario para satisfacer la misma. | Richard Duque-Herman Alarcón | 2016-04-12 | 2016-05-02 | 20 |
| Utilizar casos de usos y diagramas UML para facilitar el desarrollo | Richard Duque-Herman Alarcón | 2016-05-03 | 2016-05-30 | 27 |
| Crear diccionario de datos y Diseñar base de datos para también facilitar el desarrollo. | Richard Duque-Herman Alarcón | 2016-06-01 | 2016-06-10 | 9 |
| Realizar un estudio sobre arquitectura de software e implementar dicha arquitectura | | | | |
| Actividad | Responsable | Fecha inicio | Fecha Fin | Días |
| Buscar en internet, libros y preguntando a otros programadores como montar una buena arquitectura de software. | Richard Duque-Herman Alarcón | 2016-06-11 | 2016-06-14 | 3 |
| Investigar cuándo y por qué se deben usar 3 o más capas en la arquitectura del proyecto | Richard Duque-Herman Alarcón | 2016-06-14 | 2016-06-17 | 3 |
| Implementar arquitectura de software para iniciar un buen desarrollo ordenado. | Richard Duque- | 2016-06-18 | 2016-06-30 | 12 |

| | | | | |
|---|------------------------------|--------------|------------|------|
| | Herman Alarcón | | | |
| Diseñar prototipo de software para un fácil manejo basándonos en encuestas. | | | | |
| Actividad | Responsable | Fecha inicio | Fecha Fin | Días |
| Buscar información sobre cómo posicionar los componentes del diseño en cada uno de los formularios de acuerdo a una media. | Richard Duque-Herman Alarcón | 2016-07-01 | 2016-07-05 | 4 |
| Estudiar e implementar funcionalidades atractivas para los usuarios, por ejemplo accesos directos | Richard Duque-Herman Alarcón | 2016-07-06 | 2016-07-08 | 2 |
| Antes de empezar con el diseño del software se tiene de tener en cuenta que debe de ser de fácil uso para el usuario, ya que cualquier funcionamiento complejo puede causar la inutilidad del software. | Richard Duque-Herman Alarcón | 2016-07-10 | 2016-07-15 | 5 |
| Realizar encuestas al usuario para verificar si el diseño es intuitivo. | Herman Alarcón | 2016-07-16 | 2016-07-20 | 4 |
| Iniciar Diseño del aplicativo | Richard Duque | 2016-07-23 | 2016-07-30 | 7 |
| Codificar de acuerdo a lo encontrado en el objetivo 1. | | | | |
| Actividad | Responsable | Fecha inicio | Fecha Fin | Días |
| Montar arquitectura | Richard Duque | 2016-08-04 | 2016-08-10 | 6 |
| Implementar nomenclatura que se va a utilizar en la declaración de variables, objetos, campos, tablas, base de datos para llevar un orden durante el desarrollo. | Richard Duque-Herman Alarcón | 2016-08-12 | 2016-08-14 | 2 |
| Iniciar el desarrollo UCP-PIM | Richard Duque-Herman Alarcón | 2016-08-16 | 2016-10-30 | 75 |
| Realizar pruebas al software para garantizar su funcionamiento. | | | | |
| Actividad | Responsable | Fecha inicio | Fecha Fin | Días |

| | | | | |
|---|------------------------------|---------------------|------------------|-------------|
| Investigar sobre un buen método para realizar pruebas a un software, teniendo en cuenta la calidad. | Richard Duque-Herman Alarcón | 2016-10-01 | 2016-10-03 | 2 |
| Realizar pruebas, búsqueda de errores y corrección de los mismos. | Richard Duque-Herman Alarcón | 2016-10-03 | 2016-10-08 | 5 |
| Implantar versión beta en GooglePlayStore | | | | |
| Actividad | Responsable | Fecha inicio | Fecha Fin | Días |
| Consultar sobre como publicar un aplicativo móvil en Google | Richard Duque-Herman Alarcón | 2016-10-08 | 2016-10-12 | 4 |
| Publicar versión Beta | Richard Duque-Herman Alarcón | 2016-10-12 | 2016-10-14 | 2 |

Tabla 27 - Cronograma

3.4. PRESUPUESTO

| Concepto | Cantidad | Valor Unitario | Valor Total | Descripción |
|---------------------|----------|----------------|-------------|--|
| Computador Portátil | 2 | \$1.800.000 | \$3.600.000 | Computador portátil se utilizará durante 12 meses(Duración del proyecto) |
| Internet(Meses) | 24 | \$32.000 | \$768.000 | Se necesitara internet para la búsqueda de información en cada una de las etapas |
| Energía(Meses) | 24 | \$80.000 | \$1.920.000 | En cada una de las etapas se consumirá energía extra |
| Sillas | 2 | \$60.000 | \$120.000 | Se utilizaran sillas para comodidad a la hora de desarrollar |
| Mesas | 2 | \$100.000 | \$200.000 | Se utilizarán mesas para comodidad a la hora de desarrollar |
| Gasolina | 12 | \$10.000 | \$120.000 | El transporte será en una moto. |
| Memorias USB | 5 | \$18.000 | \$90.000 | Se utilizaran 4 memorias USB para el almacenamiento de información |
| HDD | 1 | \$80.000 | \$80.000 | Se utilizará un HDD para realizar backup y almacenar toda la información acerca del proyecto |

| | | | | |
|--------------|---|--------------|--------------|--|
| Mano de obra | 1 | \$12.000.000 | \$12.000.000 | La mano de obra valorada en dinero para este software fue de 12 millones |
| Imprevistos | 1 | \$1.000.000 | \$1.000.000 | En los imprevistos se tuvo en cuenta Papelería extra, gasolina extra, mantenimiento de computador portátil, energía extra y otros. |
| Papelería | 1 | \$200.000 | \$200.000 | Impresiones, Compra de cd's, lápices, hojas, entre otros. |
| VPS(Meses) | 6 | \$85.000 | \$510.000 | 6 meses de hosting para montar versión web del software y base de datos. |
| Total | | | \$20.608.000 | Un total de Veinte millones seiscientos ocho mil pesos colombianos |

Tabla 28 - Presupuesto

RECOMENDACIONES

Centralizar los sistemas de información ya que esto podría facilitar la manipulación de tanta información que maneja la Universidad.

Utilizar procedimientos almacenados porque estos brindan seguridad y mantenibilidad a los sistemas.

Tener documentación como diccionario de datos, modelos relacionales y una nomenclatura fija para la creación de tablas, campos, vistas, procedimientos, funciones entre otros.

No utilizar muchas tablas temporales porque estas consumen mucha memoria en el servidor y más en motores tan robustos como Oracle y PostgreSQL.

Al nivel de WebService implementar uno que sea dinámico para conectarse a diferentes bases de datos pues esto facilitará la migración de bases de datos.

Utilizar Tokens en los webservices que se desarrollen a futuro en la Universidad.

No almacenar cadenas de conexión en el archivo web.config en el caso si el webservice está construido en .Net ya que XML es un lenguaje abierto y sería fácil extraer código desde dicho archivo.

En caso de que el aplicativo móvil se le cambie el webservice al que apunta se debe modificar el archivo Metodos.Java de la carpeta webservice.

En caso de que se cambie el nombre de los atributos de las entidades del webservice se debe cambiar también en código del aplicativo móvil.

CONCLUSIONES

Se aplicó ingeniería de software y definir una buena arquitectura a todo desarrollo que se vaya a realizar ya que por medio de esta se facilitó la programación, el software quedó con más calidad y su mantenibilidad fue mejor.

Se tuvo claro el diseño de vistas del sistema a desarrollar porque es con lo que el usuario interactuará, además tener las vistas claras facilitó el desarrollo.

Se realizaron pruebas funcionales al software basadas en los requerimientos levantados, además se salió un poco de lo funcional y se trató de tumbar el sistema ya que cuando se realizan pruebas es para que el sistema falle.

APÉNDICES:

APÉNDICE A:

Manual de usuario Ver documento Manual_Usuario_UCP_PIM_2016.docx en el medio magnético.

APÉNDICE B:

Documento de pruebas: Ver PruebasUnitarias_UCP_PIM_2016.xls en el medio magnético.

BIBLIOGRAFÍA

- Abraham Silberschatz, H. F. (2002). *FUNDAMENTOS DE BASES DE DATOS*. Madrid: McGraw-Hill.
- Agudelo, F. A. (2011). La Ingeniería de Software en el desarrollo de Mobile. 5.
- Alonso, S. (Marzo de 2015). *elladodelmal*. Obtenido de <http://www.elladodelmal.com/2015/03/cuantas-apps-nuevas-hay-en-google-play.html>
- Android-sdk. (2016). *Android*. Obtenido de <http://android-sdk.uptodown.com/>
- AngelFQC, M. p. (2016). *developer.mozilla*. Obtenido de <https://developer.mozilla.org/es/docs/Web/HTML>
- Bahit, E. (2016). *duea*. Obtenido de http://www.duea.umss.edu.bo/documentos/Plan_car_184799.pdf:
http://www.duea.umss.edu.bo/documentos/Plan_car_184799.pdf
- Bahit, E. (2016). *duea*. Obtenido de http://www.duea.umss.edu.bo/documentos/Plan_car_184799.pdf:
http://www.duea.umss.edu.bo/documentos/Plan_car_184799.pdf
- Cochran, D. (12 de 11 de 2012). *Twitter Bootstrap Web Developmen*. Packt Publishing.
- Cornejo, J. E. (1 de 2008). *El Lenguaje de Modelado Unificado*. Obtenido de El Lenguaje de Modelado Unificado: <http://www.docirs.com/uml.htm>
- coverpages. (25 de 6 de 2015). *coverpages*. Obtenido de <http://xml.coverpages.org/xml.html>
- culturacion. (2012). *culturacion*. Obtenido de <http://culturacion.com/que-es-y-para-que-sirve-un-web-service/>
- developer.android. (2016). *AndroidStudio OverView*. Obtenido de <http://developer.android.com/intl/es/tools/studio/index.html>
- EL MUNDO. (11 de 2016). *elmundo.es*. Obtenido de <http://www.elmundo.es/tecnologia/2016/11/04/581c9b64e5fdea8b0e8b45c6.html>
- Elvisor. (17 de 11 de 2015). *es.wikipedia.org*. Obtenido de [es.wikipedia.org](http://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas):
http://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas
- Gestión TI y SI. (2015). Obtenido de <https://play.google.com/store/apps/details?id=co.edu.utp.mb&hl=es>
- GIT. (2016). *git-scm*. Obtenido de <https://git-scm.com/book/es/v1/Empezando-Fundamentos-de-Git>

google. (05 de 2015). *github*. Obtenido de <https://github.com/google/gson>

guru99. (2013). *guru99*. Obtenido de <http://www.guru99.com/restful-web-services.html>

Ian Sommerville Pearson. (2005). *Ingeniería de software*. Madrid: Miguel martin romo. Obtenido de Google Books: http://es.slideshare.net/jasc_584/ingenieriadesoftware-iansommerville7maedicion-9417118

James Gosling, B. J. (13 de 2 de 2015). *docs.oracle*. Obtenido de <http://docs.oracle.com/javase/specs/>

Kaspersky, E. (7 de 5 de 2013). *blog.kaspersky*. Obtenido de <https://blog.kaspersky.com.mx/explicativo-certificados-digitales-y-https/615/>

Laboda, X., & Josep Galimany, R. M. (1985). *Biblioteca práctica de la computación*. Barcelona: Ediciones Océano-Éxito.

Mateu, C. (2004). *Desarrollo de aplicaciones web*. Barcelona: UOC.

Microsoft. (11 de 2007). *msdn.microsoft*. Obtenido de [https://msdn.microsoft.com/es-co/library/6x6bk1f4\(v=vs.90\).aspx](https://msdn.microsoft.com/es-co/library/6x6bk1f4(v=vs.90).aspx)

Microsoft. (11 de 2007). *msdn.microsoft*. Obtenido de [https://msdn.microsoft.com/es-es/library/ms181238\(v=vs.90\).aspx](https://msdn.microsoft.com/es-es/library/ms181238(v=vs.90).aspx)

Microsoft. (11 de 2007). *msdn.microsoft*. Obtenido de [https://msdn.microsoft.com/es-es/library/4w3ex9c2\(v=vs.100\).aspx](https://msdn.microsoft.com/es-es/library/4w3ex9c2(v=vs.100).aspx)

Microsoft. (9 de 2008). *Technet.Microsoft*. Obtenido de [https://technet.microsoft.com/es-es/library/cc753433\(v=ws.10\).aspx](https://technet.microsoft.com/es-es/library/cc753433(v=ws.10).aspx)

Microsoft. (2016). *Msdn.Microsft*. Obtenido de <https://msdn.microsoft.com/es-co/library/kx37x362.aspx>

Microsoft. (5 de 2016). *msdn.microsoft*. Obtenido de [https://msdn.microsoft.com/es-es/library/ms731082\(v=vs.110\).aspx](https://msdn.microsoft.com/es-es/library/ms731082(v=vs.110).aspx)

Microsoft. (2016). *msdn.microsoft.com*. Obtenido de [msdn.microsoft.com: https://msdn.microsoft.com/es-es/library/hh425099%28v=vs.110%29.aspx](https://msdn.microsoft.com/es-es/library/hh425099%28v=vs.110%29.aspx)

OC Developers. (2014). *PlayStore*. Obtenido de <https://play.google.com/store/apps/details?id=com.oc.ungps>

Oracle. (7 de 2007). *oracle*. Obtenido de <http://www.oracle.com/technetwork/es/documentation/317491-esa.pdf>

Oracle. (5 de 2015). *Oracle DataBase*. Obtenido de <https://shop.oracle.com/pls/ostore/product?p1=OracleDatabase&p2=&p3=&p4=&p5=>

PhoneArena. (07 de 02 de 2012). *androidpit*. Obtenido de androidpit:
<http://www.androidpit.es/historia-app-store-smartphones>

powerdata. (7 de 5 de 2015). *blog.powerdata.es*. Obtenido de <http://blog.powerdata.es/el-valor-de-la-gestion-de-datos/bid/403485/Libros-en-pdf-sobre-normalizaci-n-de-bases-de-datos>

Priori, B. (2016). *Play Store*. Obtenido de
https://play.google.com/store/apps/details?id=com.conduit.app_498a85e366b24af4b92413faa45f4cdc.app

Raizlabs. (09 de 2015). *github*. Obtenido de <https://github.com/Raizlabs/DBFlow>

Revelo, J. (22 de 2 de 2015). *hermosaprogramacion*. Obtenido de
<http://www.hermosaprogramacion.com/2015/02/android-volley-peticiones>

searchdatacenter. (01 de 2015). Obtenido de
<http://searchdatacenter.techtarget.com/es/definicion/SQL-Server>

Ser, J. (2015). *Play Store*. Obtenido de <https://play.google.com/store/apps/details?id=com.Cidca>

SQLite. (2016). *Sqlite*. Obtenido de <https://www.sqlite.org/>

square. (2014). *github*. Obtenido de <https://github.com/square/retrofit>

TaperBlog. (19 de 05 de 2015). *TaperBlog* . Obtenido de TaperBlog :
<http://experienciadeusuario.tumblr.com/post/86217467970/que-es-bootstrap-3-y-que-ventajas-tiene-trabajar>

UCP. (2016). *ucp.edu.co*. Obtenido de <http://www.ucp.edu.co/institucion/>

Universidad Libre Seccional Barranquilla. (2015). *Play Store*. Obtenido de
https://play.google.com/store/apps/details?id=com.unilibre&hl=fr_CA

Zunino, H. G. (2006). Seguridad en web services. *Depto De Computación*.