

**DESARROLLO DEL APLICATIVO MÓVIL SOON PARA LA UNIVERSIDAD
COOPERATIVA DE COLOMBIA SEDE BOGOTÁ**

Jonathan Alexander Hernández Ruiz

Sergio Alejandro Higuera Rojas

UNIVERSIDAD COOPERATIVA DE COLOMBIA

FACULTAD DE INGENIERÍA

PROGRAMA TECNOLOGÍA EN SISTEMAS

BOGOTÁ D.C.

2017

**DESARROLLO DEL APLICATIVO MÓVIL SOON PARA LA UNIVERSIDAD
COOPERATIVA DE COLOMBIA SEDE BOGOTÁ**

**Trabajo de grado en Modalidad de Análisis Sistemático de literatura
Para optar por el título de
Tecnólogo en Sistemas**

Director

Ing. Jaime Páez.

**Línea de investigación
Desarrollo de Software**

**UNIVERSIDAD COOPERATIVA DE COLOMBIA
FACULTAD DE INGENIERÍA
PROGRAMA TECNOLOGÍA EN SISTEMAS
BOGOTÁ D.C.**

2017

Nota de aceptación:

Firma del presidente del jurado

Firma del jurado

Firma del jurado

TABLA DE CONTENIDO

1. Introducción.....	9
2. Marco referencial.....	14
2.1. Marco teórico.....	14
2.2. Marco conceptual.....	19
2.3. Marco legal.....	20
3. Desarrollo ingenieril.....	22
3.1. Metodología Ágil.....	22
3.2. Patrones.....	52
3.3. Casos de uso.....	53
3.3.1. Diagrama de casos de uso.....	53
3.3.2. Caso de uso 1-Consultar informacion de bloques.....	54
3.3.3. Caso de uso 2-Consultar ubicación de bloques.....	55
3.3.4. Caso de uso 3-Consultar informacion de programas.....	58
3.3.5. Caso de uso 4-Calcular valor matricula.....	59
3.3.6. Caso de uso 5-Realizar llamadas de emergencia.....	61
3.4. Diagrama de clases.....	64
3.5. Diagrama de componentes.....	64

3.6. Diagrama de interacción.....	65
3.7. Diagrama de secuencia.....	65
3.8. Modelo vista.....	66
3.9. Modelo de datos.....	67
3.10. Modelo relacional.....	67
3.11. Diccionario de datos.....	68
4. Conclusiones y Recomendaciones.....	69
5. Referencias bibliográficas.....	71

TABLAS

1. Plan de iteracion 1.....	28
2. Historia de usuario 1.....	29
3. Plan de iteracion 2.....	32
4. Historia de usuario 2.....	33
5. Plan de iteracion 3.....	35
6. Historia de usuario 3.....	38
7. Plan de iteracion 4.....	39
8. Historia de usuario 4.....	40
9. Plan de iteracion 5.....	42
10. Historia de usuario 5.....	43
11. Plan de iteracion 6.....	46
12. Plan de iteracion 7.....	49
13. Caso de uso 1.....	56
14. Caso de uso 2.....	58
15. Caso de uso 3.....	60
16. Caso de uso 4.....	62
17. Caso de uso 5.....	64
18. Modelo de datos.....	69
19. Diccionario de datos.....	70

GRAFICOS

1. Encuesta 1.....	11
2. Encuesta 2.....	12
3. Encuesta 3.....	12
4. Encuesta 4.....	13
5. SCRUM.....	25
6. Casos de uso.....	55
7. Caso de uso 1.....	57
8. Caso de uso 2.....	59
9. Caso de uso 3.....	61
10. Caso de uso 4.....	63
11. Caso de uso 5.....	65
12 Diagrama de clases.....	66
13. Diagrama de componentes.....	66
14. Diagrama de interaccion.....	67
15. Diagrama de secuencia.....	67
16. MVC1.....	68
17. MVC2.....	68
18.Modelo relacional.....	69

IMÁGENES

1. Paleta de colores.....	30
2. Diseño 1.....	30
3. Diseño 2.....	33
4. Diseño 3.....	36
5. Diseño 4.....	40
6. Diseño 5.....	44
7. Pruebas 1.....	51
8. Pruebas 2.....	52
9. Pruebas 3.....	53
10. Pruebas 4.....	54

1. INTRODUCCIÓN

La Universidad Cooperativa de Colombia sede Bogotá cuenta con 35 bloques distribuidos por la ciudad de Bogotá zona centro, sur, norte, Teusaquillo en los que se imparte 24 programas académicos y se encuentran inscritos en la actualidad más de 12000 estudiantes regulares. Además la Universidad facilita sus instalaciones para llevar a cabo diferentes actividades académicas como: exámenes de estado, exámenes saber Pro, exámenes de suficiencia para ingreso al estado, exámenes de ingreso a la Universidad Nacional como también cursos de extensión y capacitación para las diferentes comunidades que apoya la universidad.

El problema de investigación de esta monografía se expresa en la siguiente pregunta: ¿Conoce usted toda la información de los bloques de la Universidad Cooperativa sede Bogotá? En ese sentido, la hipótesis para responder dicha pregunta se basó en encuestas realizadas a los estudiantes de esa sede, en donde se evidencia un total desconocimiento de dicha información y se identifica una necesidad de una herramienta que pueda aportar a dicho desconocimiento.

Luego de aplicar el análisis de estas encuestas se llegó a la conclusión de realizar una aplicación para dispositivos móviles que supiera esa necesidad para mantener informado al estudiante sobre los distintos bloques de la Universidad Cooperativa en Bogotá, y no solo eso sino que además se agregó en la encuesta un par de preguntas para saber si al estudiante le gustaría que dicha aplicación tuviera un módulo para realizar el cálculo del valor de los créditos y otro para realizar llamadas de emergencia, en lo que más del 90% de los encuestados estuvieron de acuerdo. Por lo tanto se realizó el desarrollo de la aplicación Móvil SOON implementando un módulo con la información e ubicación de los bloques, otro con el cálculo de créditos de las carreras ofertadas en la Universidad Cooperativa de Colombia sede Bogotá y por ultimo un botón SOS para realizar llamadas en caso de emergencia.

-Planteamiento del problema

Los estudiantes de primer semestre que ingresan a la universidad desconocen toda su distribución y ubicación de los bloques salones y laboratorios asignados para sus clases. La universidad no tiene herramientas que faciliten información instantánea y pronta para permitir esta información, sobre todo que estas actividades se desarrollan los días sábados y domingos.

Una vez finaliza el semestre es normal que el estudiante desee llevar a cabo el cálculo del valor de la matrícula para la inscripción de materias del próximo semestre, y muchas veces se desconoce el valor del crédito debido a un aumento lo que lleva a que realice una búsqueda de este para calcular el valor total de la matrícula respecto a la cantidad de créditos.

Se han presentado casos de emergencia como desmayos, epilepsias y otras enfermedades de la comunidad académica y no se cuenta con un recurso instantáneo para ubicar llamadas a las diferentes entidades que prestan sus servicios a la Universidad para acudir a emergencias.

-Objetivos

Objetivo general

Desarrollar el aplicativo móvil SOON para la universidad Cooperativa de Colombia sede Bogotá, compuesto de los módulos ubicación de bloques, calculadora valor matrícula, y botón de SOS.

Objetivos específicos

- Analizar la información de los bloques de la Seccional Bogotá de la Universidad Cooperativa de Colombia.
- Diseñar los esquemas y patrones para el aplicativo móvil SOON. Bajo la metodología SCRUM.
- Desarrollar el aplicativo móvil SOON.
- Implementar el aplicativo móvil SOON.
- Realizar pruebas sistemáticas para la validación del correcto funcionamiento de la aplicación.

-Justificación

Colombia es uno de los países con gran penetración de la telefonía móvil, las estadísticas indican que hay más de 55 millones de líneas móviles, si tenemos en cuenta esto podemos evidenciar que tener un dispositivo móvil en la actualidad ya es casi una necesidad en base a que no solo sirve para realizar llamadas sino que también tiene otras funcionalidades, como lo es tomar fotografías, realizar videos, organizar archivos, entre otras en las que podemos destacar la instalación y uso de aplicaciones.

Se evidencia en la Universidad, la necesidad de consulta de la información por la cantidad de bloques y distancia, así como los distintos servicios que se prestan en cada una, debido a que no existe una herramienta de apoyo para el estudiante ni tampoco una presentación de las sedes solo un almanaque de direcciones.

Las aplicaciones móviles además de ser una herramienta sirven de publicidad para generar el ingreso los nuevos estudiantes que se pueden ver atraídos por la información de las distintas sedes, y más si tenemos en cuenta que la Universidad Cooperativa de Colombia es la universidad con mayor cantidad de sedes en el país.

No existe una herramienta sistemática, para que los estudiantes tengan la oportunidad de ver las instalaciones que componen la Universidad, de hecho, solo se hace una inducción en la sede en la que deben de asistir la primera semana, en cuanto a las demás se ven en la necesidad de informarse.

Se realizó una encuesta a 50 estudiantes de diferentes semestres y carreras donde se pudo evidenciar la siguiente información:

De la pregunta, ¿Ha tenido dificultades a la hora de ubicar alguno de los bloques? 3 estudiantes contestaron que no y 47 que si lo que indica que el 94% de los estudiantes encuestados han tenido alguna dificultad al momento de tener que ubicar alguno de los bloques de la universidad.

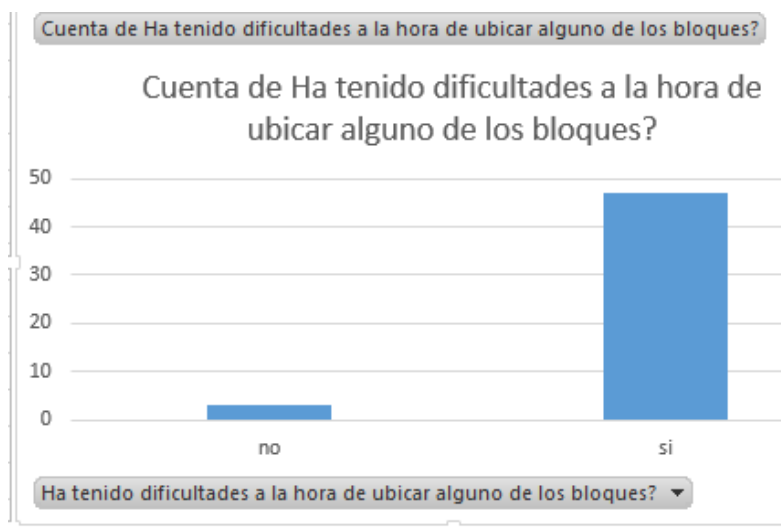


Grafico 1. Encuesta 1 elaborado por Autores

De la pregunta, ¿Sabe usted con cuantos bloques cuenta la Universidad Cooperativa de Colombia en Bogotá? 49 estudiantes contestaron que no y 1 que si lo que indica que el 98% de los encuestados desconocen la cantidad de bloques en la ciudad de Bogotá.

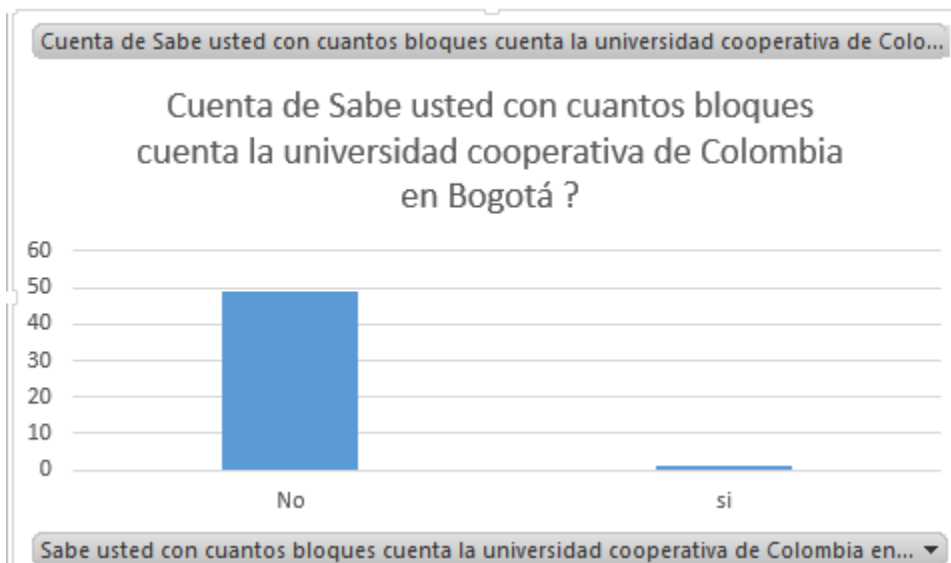


Grafico 2 Encuesta 2 elaborado por Autores

De la pregunta ¿Le gustaría que se implementara una aplicación que le ayudara con la información de todos y cada uno de los bloques? 47 estudiantes contestaron que sí y 3 no, lo que indica que el 94% de los encuestados considera que implementar una aplicación con esa información sería de gran ayuda, y el otro 6% considera que no, porque ya están cursando sus últimos semestres.

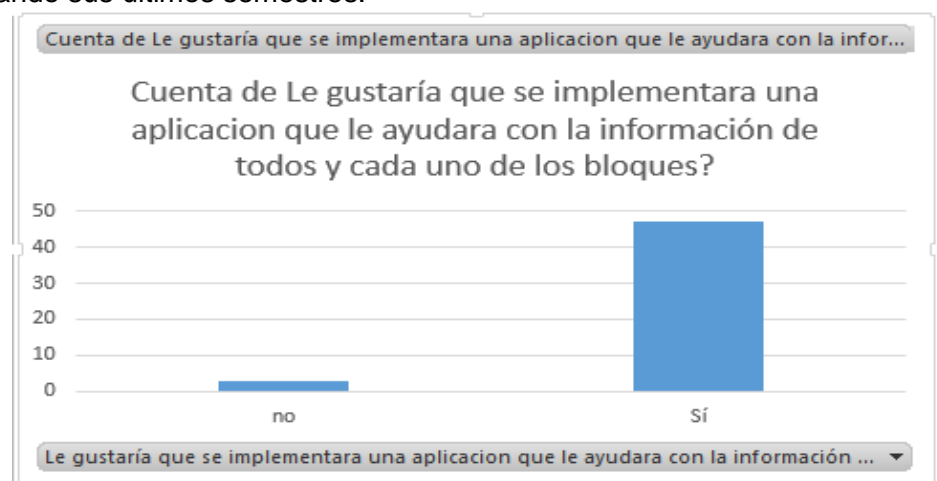


Grafico 3 Encuesta 3 elaborado por Autores

De la pregunta ¿Le gustaría que esta aplicación además contara con el cálculo del costo del próximo semestre? 47 estudiantes contestaron que sí y 3 no, lo que indica que el 94% de los encuestados considera que implementar ese modulo a la aplicación sería de gran utilidad, y el otro 6% considera que no, porque ya están cursando sus últimos semestres.

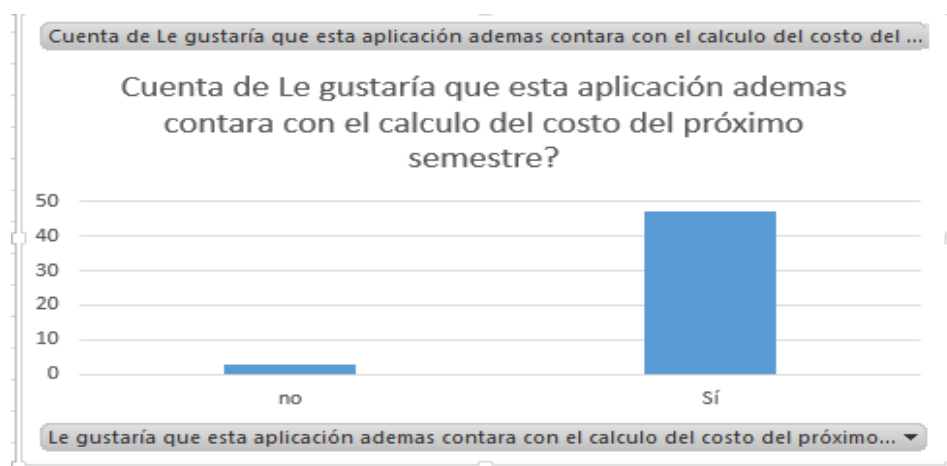


Grafico 4 Encuesta 4 elaborado por Autores

Con este análisis podemos concluir que de cada 50 estudiantes encuestados más del 90% no tiene a su alcance información sobre los distintos bloques de la Universidad sede Bogotá y están de acuerdo con la implementación de una herramienta que pueda ayudarles con eso adicionando el módulo de los créditos.

2. MARCO REFERENCIAL

2.1. MARCO TEÓRICO

Para llevar a cabo el diseño y desarrollo de una aplicación para dispositivos móviles se deben tener en cuenta los siguientes pasos:

Tener una buena idea

Si ya tienes una idea para tu aplicación, puedes saltarte este punto. Si no es así, lo que realmente necesitas es buscar un problema o necesidad al que darle solución, y estos están por todas partes. Los empresarios que han tenido éxito en el desarrollo de Apps móviles han sido capaces de resolver problemas de una manera simple. Cuando miras a tu alrededor, cada producto y servicio que utilizas fueron creados para resolver un problema o cubrir una necesidad. Así que trata de buscar problemas o necesidades en tu vida diaria y realiza una lista. Una vez que tengas una lista exhaustiva, ya puedes empezar a pensar en cómo puedes resolverlos.

Definir el objetivo de la aplicación

Tener una gran idea o encontrar un punto del mercado al que no esté dando un producto o servicio es el punto de partida en cada nuevo proyecto. Antes de ir directamente a detallar embargo, debe definir claramente el propósito y la misión de la aplicación móvil:

¿Qué va a hacer la aplicación?

¿Cuál es su atractivo principal? Fidelización, uso recurrente (es decir que hay contenido dinámico por ejemplo) o uso único (imagina una app que sólo vale para hacer una felicitación de navidad, pues lo normal es que después de navidad el usuario la desinstale). Este punto nos va a marcar el análisis a posteriori que tengamos que hacer para saber si es normal las tasas de desinstalación, recurrencia de acciones...etc

¿Qué problema concreto va a resolver? ¿Qué necesidad va a cubrir?

Nota: Definir un objetivo claro para la aplicación también va a ayudar a que su desarrollo sea más barato y rápido.

Investigar la posible demanda de la aplicación

Realizar una investigación en el mercado objetivo demostrará que existe una demanda para la aplicación y que no eres el único que lo considera necesario. Una de las posibles maneras para validar tu idea es utilizar la herramienta Google Keyword Planner, analizando el

número de usuarios que buscan solución a ese problema. También puedes crear una landing page que detalle la idea de la aplicación y dar la opción de ampliar la información a los usuarios que lo soliciten por correo electrónico.

Otra posibilidad es analizar las tiendas de aplicaciones. Analiza qué aplicaciones tienen éxito y qué tipo de problemas resuelven. Muchas veces podemos encontrar aplicaciones que dan solución a un problema pero no de la mejor manera, y sin embargo tienen éxito por ser la única o de las pocas existentes. En ese caso debes plantearte si la aplicación a desarrollar es capaz de mejorar la ya existente y abarcar ese mercado.

Ejemplo: Un ejemplo de esto es Emoji. Se realizó un análisis de las aplicaciones de emoticonos existentes que tenían éxito y se comprobó que había una falta de variedad y funcionalidad en todas ellas, lo que le daba un amplio margen de mejora. En sólo dos semanas desarrollaron una aplicación con más emoticonos que podía ser utilizada en SMS, correo electrónico, Facebook, etc.. y se situó en apenas 6 días en el primer puesto de su categoría y puesto 12 en la clasificación general, con una recaudación de \$500/día a través de un sistema Freemium.

Otra opción es el gran mercado para los diseñadores llamado Dribbble. Los diseñadores utilizan Dribbble para mostrar su trabajo a los demás, recibir retroalimentación y para conseguir la inspiración de otros artistas. Es un lugar muy recomendable para buscar ideas sobre diseños e implementaciones.

Registrarse como desarrollador o busca uno

Una vez que tengas una idea clara y que sabes que hay demanda para la aplicación móvil, es necesario registrarse como desarrollador en aquella plataforma en la que quieras desarrollar tu app, ya sea Android, iOS o Windows Phone. No te dejes intimidar por la palabra “desarrollador”, no significa que tengas que ser el programador, es simplemente el nombre que se utiliza para la persona encargada de publicar las aplicaciones.

Dependiendo del objetivo de la App va a ser mejor desarrollarla en Apple iOS o Google Android (en una de las dos), ya que vas a estar en tiendas de aplicaciones con mucha demanda (también mucha competencia, que no se te olvide). Por ejemplo, los usuarios de iOS son más propensos a hacer compras en apps o pagar por descargarse una aplicación. Además depende del mercado dónde lo queramos posicionar tendrá más sentido crearla en Android o en IOS (hay que analizar el volumen de dispositivos por sistema operativo).

Nota: Apple establece una normativa muy severa para todas las aplicaciones que, de no ser cumplida tu aplicación, tu app será rechazada.

La contratación del desarrollador puede ser un proceso largo, pero es un tiempo bien empleado ya que de él va a depender en gran parte el éxito del proyecto. Hacer una buena

elección te evitará retrasos innecesarios, costes innecesarios y frustración en el futuro. Analiza sus anteriores trabajos para ver sus habilidades a la hora de plasmar las ideas en una aplicación. Comprueba que comprende bien tu idea y que es capaz de desarrollarla en su totalidad. También puedes acudir a empresas especializadas que cuentan con varios desarrolladores y una amplia experiencia para asegurarte unos mejores resultados.

Crear bocetos de la aplicación

Desarrollando bocetos sentarás las bases para la futura interfaz. En este paso tienes que conceptualizar visualmente las principales características, el trazado aproximado y la estructura de la aplicación. Tener un primer esbozo de la aplicación ayudará a todo el equipo a comprender el objetivo de la app, ya que se van a utilizar como referencia para las próximas fases del proyecto.

Ya sea con papel y lápiz o utilizando medios digitales como Photoshop, lo importante es dar forma a lo que tienes en la cabeza de manera que un programador pueda empezar a entender lo que tratas de crear. Para este paso resulta de utilidad fijarse en otras aplicaciones ya existentes que sirvan como inspiración y modelo del que podamos extraer ideas. De esta manera facilitaremos la comunicación con el programador.

Crear el Wireframe y el Storyboard

En esta fase las ideas y las características principales de la app se van viendo mucho más claras. Wireframing es el proceso de crear una maqueta o prototipo de la aplicación. Existen muchas herramientas online para crear de prototipos de apps. Las más populares son Balsamiq, Moqups, y HotGloo, que te permiten colocar y estructurar todos los datos y gráficos representativos en su lugar, además de añadir funcionalidad con la colocación de botones, de modo que puedes navegar a través de tu aplicación.

Mientras que estás trabajando en los wireframes también tienes que empezar a crear el storyboard o guión gráfico de la aplicación. La idea es construir una hoja de ruta que te ayudará a entender la conexión entre cada pantalla y cómo el usuario navegará a través de la aplicación.

Definir el back end

Los wireframes y storyboard ahora se convierten en la base de la estructura del back end. Dibuja un boceto de los servidores, APIs y diagramas de datos. Esta será una referencia muy útil para el desarrollador, y a medida que más personas se unan al proyecto tendrás un diagrama que explica toda la aplicación, para que no tengas que hacerlo tu desde el principio.

Nota: Crea tus wireframes y storyboard de acuerdo con las limitaciones técnicas, si los hay.

Crear un prototipo

Usa tus wireframes para pedir opinión sobre la aplicación a amigos, familiares, colegas y expertos para crear un primer prototipo. Si es posible, invítales a tu estudio y haz que prueben el prototipo en frente de ti y del desarrollador jefe. Dale acceso al wireframe y deja que prueben la app a fondo. No dejes que se vayan sin que te den una opinión honesta, que te ayudará a identificar fallos o enlaces sin salida. Monitoriza cómo utilizan la aplicación, tomando nota de sus acciones y adaptando la interfaz de usuario UI/UX a ellos.

Nota: El objetivo es concretar el concepto de la aplicación móvil antes de que entre en el proceso de diseño. Una vez que comiences el diseño es mucho más difícil hacer cambios, por lo que cuanto más partido saques al prototipo, mejor.

Construir el back end

Ahora que ya has definido definitivamente la aplicación, es el momento de empezar a trabajar en el back end del sistema. El desarrollador tendrá que configurar servidores, bases de datos, APIs y soluciones de almacenamiento.

Nota: Otra cosa muy importante en esta paso es registrar cuentas de desarrollador en las tiendas de aplicaciones para las que estés desarrollando tu app

Diseño de las “Skins” de la App

“Skins” son lo que los diseñadores/desarrolladores llaman a las pantallas necesarias para la aplicación. El trabajo del diseñador será crear versiones de alta resolución de lo que antes eran wireframes. En definitiva convertir bocetos y notas de papel (incluidos post-it) en archivos digitales que se puedan ver en alta definición.

Nota: En este paso es crucial incluir y dejar muy claros los comentarios y opiniones de las pruebas del prototipo (del paso #8)

Testeo de la aplicación

Una vez que el diseñador acabe de diseñar las skins o las pantallas, todo estará listo para otra ronda de pruebas. No pienses que todo está listo con los pasos que has dado hasta ahora. Por primera vez tienes ese concepto en formato real, con todos los gráficos y todo el texto insertados, lo que significa que puede empezar a testear la aplicación en su diseño previo a la publicación en la App Store. Solidify y Framer son buenas aplicaciones de

pruebas para poder probar tu aplicación. Estas aplicaciones permiten importar el diseño de la aplicación y añadir enlaces donde sea necesario para probar el flujo de pantalla a pantalla.

Nota: No confundas este paso con la creación de wireframe del paso #6. Al principio se trataba de crear los aspectos básicos de la aplicación, ahora hay que implementar el diseño real, poner enlaces y hacerlo clickable.

Revisar y mejorar la App

Una vez que has dado sometido a todas las pruebas posibles al diseño de tu app y has recogido más feedback de los futuros usuarios, debes utilizar estas nuevas ideas para pulir la idea, el uso y el objetivo de la aplicación. Todavía puede cambiar el diseño y pedir a tu desarrollador que haga cambios y mejores en el back end.

Refinar cada detalle

A medida que siga el desarrollo tendrás que tener una mirada constante y crítica en tu nueva aplicación. En Android, por ejemplo, el archivo de la aplicación es fácil de instalar en un dispositivo para probar su funcionalidad en un entorno real. iOS, sin embargo, es diferente ya que requiere una plataforma como TestFlight para descargar y probar la aplicación a medida que avances en el desarrollo.

Nota: Este es el último paso en el proceso de desarrollo de la aplicación móvil

ASO o Optimización de la App

En las App Store existen millones de apps, por lo que para lograr tener éxito es necesario optimizar la aplicación. Para ello hay que tener en cuenta una serie de aspectos que pueden influir en las descargas logradas. Escoger ciertas palabras clave, usar ciertos iconos y descripciones, realizar promoción externa al mercado de aplicaciones o contratar publicidad son algunas de las técnicas que se pueden aplicar. Aquí te dejo todas las que tienes que tener en cuenta.

Título: para optimizar la app lo mejor es aprovechar el título para incluir las palabras claves que la definan. Procura que sea lo más corto posible para que de esta manera los buscadores de los distintos dispositivos lo puedan leer al completo.

Categoría: debes de escoger una sola categoría para la aplicación, pero aunque parezca sencillo no lo es tanto. Tienes que colocarte en la piel del usuario para decidir dónde la buscarían en la app.

Descripción: se trata de captar la atención de los usuarios. Puedes aprovechar este espacio para añadir un CTA (Call to action) atractivo que despierte el interés y convenga a los dudosos para que finalmente opten por la descarga de tu aplicación.

Utilizar palabras claves: el éxito o el fracaso de la estrategia de ASO dependerá básicamente de las palabras claves que vayas a utilizar. Por eso intenta escoger siempre las más adecuadas y que estén vinculadas con las principales funciones de la app.

Icono: a la hora de tomar decisiones nos dejamos llevar por la imagen, de ahí que sea importante escoger un icono llamativo y original que permita transmitir la filosofía de la aplicación. Siempre será mejor apostar por la sencillez antes que por algo complejo. Procura realizar un análisis con respecto a la competencia para detectar si tu aplicación tiene éxito o pasa desapercibida.

Capturas de pantalla: se trata de una forma muy visual de presentar la aplicación, que puede llegar a ser decisiva para que el usuario se la acabe descargando finalmente. Procura escoger aquellas que resulten atractivas visualmente y que además describan lo mejor posible las características de la aplicación.

Publicar la aplicación en el App Store

Una vez probada la aplicación y realizados los cambios necesarios es el momento de publicar la aplicación en las App Store. Android, por ejemplo, no revisa las aplicaciones recién enviadas de inmediato. Ellos la comprobarán en algún momento para ver si funciona, pero la aplicación se publicará casi al instante en Google Play.

Al contrario, el tiempo que Apple se va a tomar para revisar y aprobar o rechazar tu aplicación dependerá de si vas a enviarlo como individuo o como una empresa. Si es un individuo, toma de tres a siete días. Si es una empresa, tendrás que esperar de siete a 10 días.

Para superar esta espera hay algo que puedes hacer: presentar la app en PreApps. PreApps es un mercado de aplicaciones que ofrece a los desarrolladores la oportunidad de llegar a “early adopters”, que son usuarios a los que les gusta probar cualquier app nueva en primer los primeros. De esta manera empezarás a feedback muy valiosos que podrá decirte si va todo por buen camino o hay que empezar a trabajar en la siguiente versión.

2.2 MARCO CONCEPTUAL

Ingeniería de software:

Análisis de requerimientos: Se extraen los requisitos del producto de software. En esta etapa la habilidad y experiencia en la ingeniería del software es crítica para reconocer

requisitos incompletos, ambiguos o contradictorios. Usualmente el cliente/usuario tiene una visión incompleta/inexacta de lo que necesita y es necesario ayudarlo para obtener la visión completa de los requerimientos. El contenido de comunicación en esta etapa es muy intenso ya que el objetivo es eliminar la ambigüedad en la medida de lo posible.

Especificación: Es la tarea de describir detalladamente el software a ser escrito, de una forma rigurosa. Se describe el comportamiento esperado del software y su interacción con los usuarios y/o otros sistemas.

Diseño y arquitectura: Determinar cómo funcionará de forma general sin entrar en detalles incorporando consideraciones de la implementación tecnológica, como el hardware, la red, etc. Consiste en el diseño de los componentes del sistema que dan respuesta a las funcionalidades descritas en la segunda etapa también conocidas como las entidades de negocio. Generalmente se realiza en base a diagramas que permitan describir las interacciones entre las entidades y su secuenciado.

Programación: Se traduce el diseño a código. Es la parte más obvia del trabajo de ingeniería de software y la primera en que se obtienen resultados “tangibles”. No necesariamente es la etapa más larga ni la más compleja aunque una especificación o diseño incompletos/ambiguos pueden exigir que, tareas propias de las etapas anteriores se tengan que realizarse en esta.

Prueba: Consiste en comprobar que el software responda/realice correctamente las tareas indicadas en la especificación. Es una buena praxis realizar pruebas a distintos niveles (por ejemplo primero a nivel unitario y después de forma integrada de cada componente) y por equipos diferenciados del de desarrollo (pruebas cruzadas entre los programadores o realizadas por un área de test independiente).

Documentación: Realización del manual de usuario, y posiblemente un manual técnico con el propósito de mantenimiento futuro y ampliaciones al sistema. Las tareas de esta etapa se inician ya en la primera fase pero sólo finalizan una vez terminadas las pruebas. Mantenimiento: En esta etapa se realizan un mantenimiento correctivo (resolver errores) y un mantenimiento evolutivo (mejorar la funcionalidades y/o dar respuesta a nuevos requisitos).

Referencia de <http://proyectosguerrilla.com/blog/2013/02/las-cinco-etapas-en-la-ingenieria-del-software/>

Programación orientada a objetos:

La programación orientada a objetos (POO, u OOP según sus siglas en inglés) es un paradigma de programación que viene a innovar la forma de obtener resultados. Los objetos manipulan los datos de entrada para la obtención de datos de salida específicos, donde cada objeto ofrece una funcionalidad especial.

Muchos de los objetos pre-diseñados de los lenguajes de programación actuales permiten la agrupación en bibliotecas o librerías, sin embargo, muchos de estos lenguajes permiten al usuario la creación de sus propias bibliotecas.

Está basada en varias técnicas, incluyendo herencia, cohesión, abstracción, polimorfismo, acoplamiento y encapsulamiento.

Su uso se popularizó a principios de la década de 1990. En la actualidad, existe una gran variedad de lenguajes de programación que soportan la orientación a objetos.

Referencia de <https://msdn.microsoft.com/es-co/library/bb972232.aspx>

Bases de datos:

Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, siendo este un componente electrónico, por tanto se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos.

Referencia de <http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>

SQLite:

Es un sistema de gestión de bases de datos relacional compatible con ACID, contenida en una relativamente pequeña (~275 kiB)² biblioteca escrita en C. SQLite es un proyecto de dominio público¹ creado por D. Richard Hipp.

A diferencia de los sistemas de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. El conjunto de la base de datos (definiciones, tablas, índices, y los propios datos), son guardados como un sólo fichero estándar en la máquina host. Este diseño simple se logra bloqueando todo el fichero de base de datos al principio de cada transacción.

Referencia de <https://www.sqlite.org/about.html>

Android Studio:

Es un nuevo entorno de desarrollo integrado para el sistema operativo Android lanzado por Google, diseñado para ofrecer nuevas herramientas para el desarrollo de aplicaciones y alternativa al entorno Eclipse, hasta ahora el IDE más utilizado.

Al crear un nuevo proyecto en Android Studio, la estructura del proyecto aparece con casi todos los archivos dentro del directorio SRC, un cambio a un sistema de generación basado Gradle que proporcionará una mayor flexibilidad para el proceso de construcción. Además, gracias a su sistema de emulación integrado, Android Studio permite ver los cambios que realizamos en nuestra aplicación en tiempo real, pudiendo además comprobar cómo se visualiza en diferentes dispositivos Android con distintas configuraciones y resoluciones de forma simultánea.

Referencia de <https://developer.android.com/studio/index.html?hl=es-419>

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 diez millones de usuarios reportados.

Referencia de <https://academy.oracle.com/es/solutions-java.html>

Adobe Photoshop

Es un editor de gráficos rasterizados desarrollado por Adobe Systems Incorporated. Usado principalmente para el retoque de fotografías y gráficos, su nombre en español significa literalmente "taller de fotos". Es líder mundial del mercado de las aplicaciones de edición de imágenes y domina este sector de tal manera que su nombre es ampliamente empleado como sinónimo para la edición de imágenes en general.

Referencia de <https://www.adobe.com/la/products/photoshop/free-trial-download.html>

UML

El lenguaje unificado de modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el Object Management Group (OMG).

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional, Rational Unified Process o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

Referencia de <https://www.lucidchart.com/pages/es/qu%C3%A9-es-el-lenguaje-unificado-de-modelado-uml>

2.3 MARCO LEGAL

- Decreto 585 del 26 de Febrero de 1991 por el cual se crea el Consejo Nacional de Ciencia y Tecnología, se reorganiza el Instituto Colombiano para el Desarrollo de la Ciencia y la Tecnología-Colciencias- y se dictan otras disposiciones.
- Decreto 584 del 26 de Febrero de 1.991, por el cual se reglamentan los viajes de estudio al exterior de los investigadores nacionales.
- Decreto 393 del 26 de Febrero de 1991 por el cual se dictan normas sobre asociación para actividades científicas y tecnológicas, proyectos de investigación y creación de tecnologías.
- Ley 29 de Febrero de 1990, por la cual se dictan disposiciones para el fomento de la investigación científica y el desarrollo tecnológico y se otorgan facultades extraordinarias.
- LEY 1266 DE 2008 - **principio informático de Habeas data.**
- 1273 de 2009 - **protección de datos.**
- Artículo 183 de la Ley 23 de 1982 - **derechos de autor - Generalidades.**
- Artículo 30 de la Ley 23 de 1982 - **derechos de autor – Derechos del autor.**
- Artículo 6º y 7º de la Ley 23 de 1982 - **derechos de autor – Fines.**
- Artículos 19, 83 y 92 de la Ley 23 de 1982 - **derechos de autor – Colaboración.**
- Artículos 18 y 82 de la Ley 23 de 1982 - **derechos de autor – Equipos de investigación.**
- Dirección nacional de derecho de autor unidad administrativa especial ministerio del interior y de justicia - manual de derecho de autor (2010) - 5.5.2 Entidades de Educación
- CIRCULAR No. 06 (abril 2012) - El derecho de autor en el ámbito universitario - Unidad Administrativa Especial Dirección Nacional de derecho Autor.
- Convenio de Berna
- Tratado de la OMPI
- Ley 565 de 2000
- Decisión Andina 351 de 1993

- Ley 33 de 1987
- y la legislación vigente aplicable.

3 DESARROLLO INGENIERIL

3.1 METODOLOGÍA ÁGIL

Para llevar a cabo el desarrollo de la aplicación se uso la metodologia SCRUM.

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

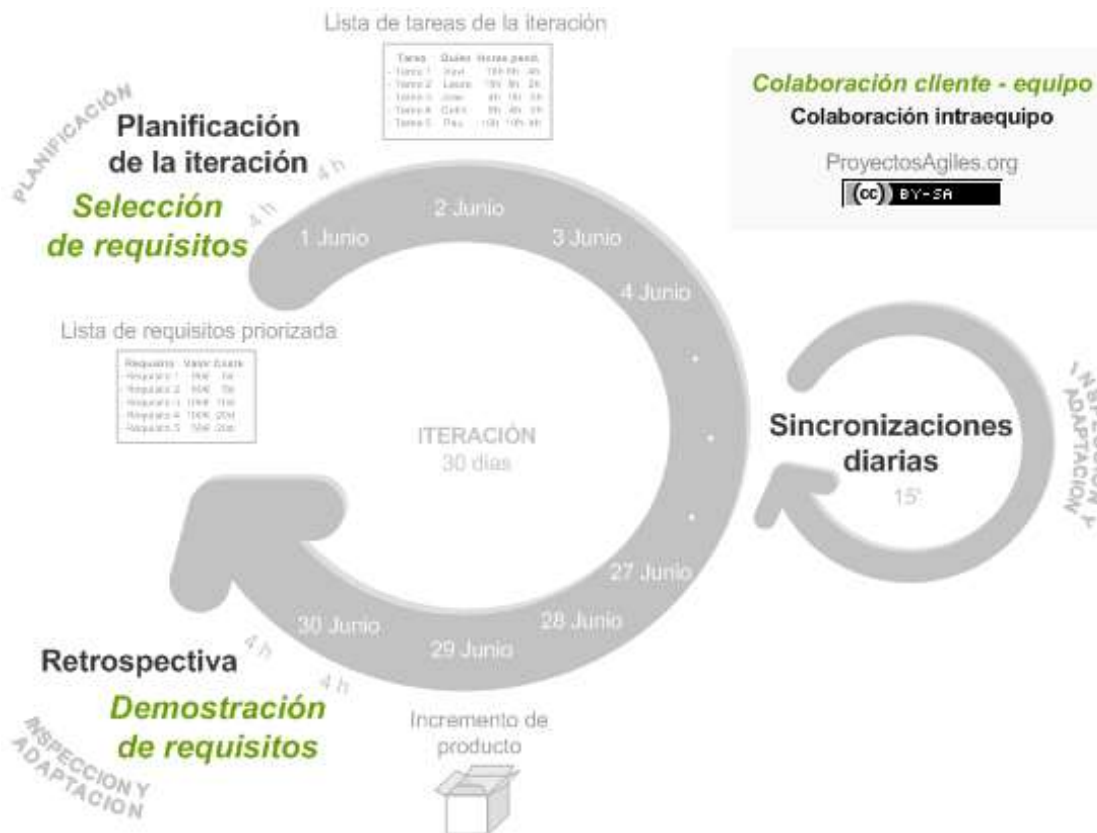


Grafico 5 SCRUM elaborado por <https://proyectosagiles.org/que-es-scrum/>

Se asignaron los respectivos roles entre los dos desarrolladores y quedo evidenciado en 14 actas y 7 planes de iteracion el desarrollo de la aplicación y sus componentes, como se evidencia a continuacion:

-ACTA 1

3 de Septiembre del 2017

Propuesta final Aplicación móvil SOON

-A partir de la investigación en todos y cada uno de los bloques de la Universidad Cooperativa de Colombia sede Bogotá, se centralizará la información más relevante en una aplicación móvil denominada con el nombre de SOON, donde el usuario podrá obtener la ubicación exacta del bloque solicitado. Además se agregaran dos módulos, uno para calcular el valor de la matricula respecto al valor del crédito de las carreras de pregrado ofertadas en Bogotá y otro modulo para realizar llamadas en caso de emergencia.

Arquitectura y tecnologías

-Entorno de Desarrollo Integrado (IDE)

-Android Studio 2.33

-Sistema de Gestión de Base de Datos (DBM)

-SQLite

-Editor de imágenes

-Photoshop

-Repositorio

-Github

Metodología

-Basado en entregas continuas

-Entrega de un mínimo producto viable

-El entregable siempre será un producto funcional para el cliente

-Duración de cada iteración: 2 semanas

-Día de entrega y revisión del entregable: Domingos 8pm

-Roles

-Analista

- Historias de Usuario
- Todo lo referente al perfil de usuario

-Diseñador Lógico

- Desarrollo ingenieril

-Diseñador Grafico

- Gráficos vistas módulos.

-Programador

- Desarrollo aplicación

-DBA

- Estructura y creación y manejo base de datos.

-Documentador y gestor de la información

- Documentación de toda la información.

-Tester

- Estructura de pruebas

-Gerente y SQA

- Planes de iteración
- Actas y aprobación de entregables

Entregables:

1. Pantalla principal.....17 Septiembre
2. Lista bloques..... 1 Octubre
3. Información por bloque.....15 Octubre
4. Mapa con ubicación por bloque.....29 Octubre
5. Lista programas y Calculadora créditos.....12 Noviembre

6. Modulo llamadas..... 26 Noviembre

7. Correcciones.....3 Diciembre

Plan de iteracion 1

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro		-Historia de usuario entrega 2
	Analista	-Paleta de colores
	Diseñador Grafico	-Diseño pantalla principal
	Tester	-Documentación de acceso a la información
	Documentador y gestor de la información	-Información bloques
Hernandez Ruiz Jonathan Alexander		Pantalla principal
	Programador	-Diagrama de clases
	Diseñador lógico	-Caso de uso 1
	DBA	-Acta 1 y 2
	Gerente y SQA	-Plan de iteración entrega 2

Tabla 1 Plan de iteracion 1 elaborado por Jonathan Hernandez

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 2

10 de Septiembre de 2017

Repositorio

- Repositorio
 - <https://github.com/Jarblack/SOON>
- Estructura de directorios
 - <https://github.com/Jarblack/SOON/tree/master/Documentacion>
- Relación con cuenta

- git config --global user.name "user_github"
- git config --global user.email "correo_github"
- Manejo básico
- <https://www.youtube.com/watch?v=bknZdA0ckHw>

Observaciones

- Herramienta de Diagramación recomendada <https://www.lucidchart.com/>

Revisión de entregables

<https://github.com/Jarblack/SOON/tree/master/Documentacion/Planes%20de%20iteracion>

Acceso de información

<https://github.com/Jarblack/SOON/tree/master/Documentacion/Gestion%20de%20la%20informacion>

-Analista

Se aprueba historia de usuario entrega 2

Historia de Usuario – SOON UCC	
Dependencia	ANALISTA
Nombre de la Historia	LISTA DE BLOQUES
Descripción	Como usuario de la aplicación móvil SOON UCC deseo poder visualizar una lista de todos los bloques de la Universidad Cooperativa sede Bogotá donde pueda consultar información actualizada de todos y cada una de ellos.
Criterios de Aceptación	Lista con todos los bloques sede Bogotá.
	En la lista debe ir una descripción de todos y cada uno de los bloques.
	En la lista debe ir una foto representativa de todos y cada uno de los bloques.
	Al seleccionar un bloque debe aparecer la respectiva información actualizada.
	En la información debe ir el número del bloque, la descripción, la dirección, transporte, características y la ubicación.

Tabla 2. Historia de usuario 1 elaborado por Autores

-Diseñador Grafico

-Se aprueba paleta de colores y diseño de pantalla principal

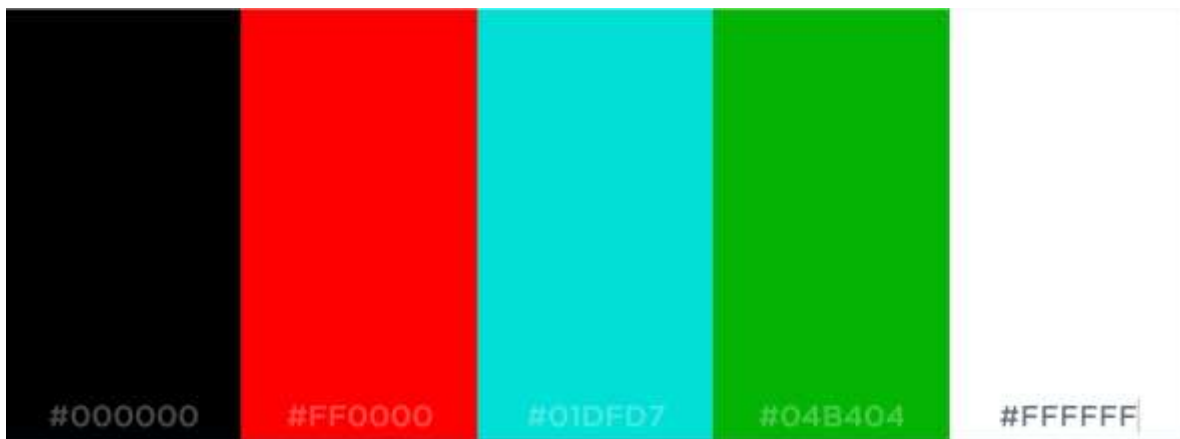


Imagen 1 Paleta de colores elaborado por Autores

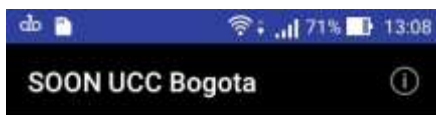


Imagen 2 Diseño 1 elaborado por Autores

-Diseñador Lógico

-Se aprueba caso de uso 1 y diagrama de clases.

Caso de uso 1 pag. 49, Diagrama de clases pag. 59

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Historia de usuario entrega 2 -Paleta de colores -Diseño pantalla principal -Documentación de acceso a la información -Información bloques
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	Pantalla principal -Diagrama de clases -Caso de uso 1 -Acta 1 y 2 -Plan de iteración entrega 2

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 3

17 de Septiembre del 2017

Documentador y gestor de la información

- Avance información bloques.

<https://github.com/Jarblack/SOON/tree/master/Documentacion/Gestion%20de%20la%20informacion>

Programador

- Desarrollo pantalla principal.

Gerente y SQA

-Acta 1 y 2 realizadas

<https://github.com/Jarblack/SOON/tree/master/Documentacion/Actas>

-Plan de iteración entrega 2 presentado.

<https://github.com/Jarblack/SOON/tree/master/Documentacion/Planes%20de%20iteracion>

-Plan de iteracion 2

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista	Historia de usuario entrega 3 -Diseño pantalla información por bloque
	Diseñador Grafico	
	Tester	
	Documentador y gestor de la información	
Hernandez Ruiz Jonathan Alexander		Información bloques
		Pantalla lista bloques
	Programador	-Diagrama de componentes
	Diseñador lógico	
	DBA	-Acta 3 y 4
	Gerente y SQA	Plan de iteración entrega 3

Tabla 3 Plan de iteracion 2 elaborado por Autores

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 4

24 de Septiembre del 2017

Analista

-Se aprueba historia de usuario

Historia de Usuario – SOON UCC	
Dependencia	ANALISTA
Nombre de la Historia	UBICACIÓN BLOQUES
Descripción	Como usuario de la aplicación móvil SOON UCC deseo poder ubicar mediante un mapa el bloque seleccionado de la lista de bloques.
Criterios de Aceptación	En la sección de ubicación de la información del bloque debe haber un botón que muestre el mapa con la ubicación del bloque.
	Mapa con ubicación de todos y cada uno de los bloques.
	Enlazarlo a Google <u>Maps</u> para que arroje medios de transporte para llegar al sitio

Tabla 4 Historia de usuario 2 elaborado por Autores

Diseñador Grafico

-Diseño de pantalla lista bloques



Imagen 3 Diseño 2 elaborado por Autores

Diseñador lógico

-Diagrama de componentes aprobado

Diagrama de componentes pag. 59

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista	Historia de usuario entrega 4
	Diseñador Grafico	-Diseño pantalla información por bloque
Hernandez Ruiz Jonathan Alexander	Tester	Información bloques
	Documentador y gestor de la información	Pantalla lista bloques
	Programador	-Diagrama de componentes
	Diseñador lógico	-Acta 3 y 4
	DBA	Plan de iteración entrega 3
	Gerente y SQA	

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 5

01 de Octubre del 2017

Documentador y gestor de la información

-Aprobada información de bloques, continuar con la investigación

Programador

-Aprobada pantalla lista bloques

Gerente y SQA

-Realizada Acta 3 y 4. Plan de iteración 3

-Plan de iteracion 3

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista	-Primer Historia de usuario entrega 5
	Diseñador Grafico	-Diseño pantalla lista programas
Hernandez Ruiz Jonathan Alexander	Tester	
	Documentador y gestor de la información	Información bloques
		-Pantalla información por bloque
	Programador	-Diagrama de interacción
	Diseñador lógico	-Caso de uso 2
	DBA	-Acta 4 y 5
	Gerente y SQA	Plan de iteración entrega 4

Tabla 5 Plan de iteracion 3 elaborado por Autores

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 6

08 de Octubre del 2017

Diseñador grafico

-Aprobado diseño pantalla lista carreras



Imagen 4 Diseño 3 elaborado por Autores

Diseñador lógico

-Aprobado diagrama de interacción y caso de uso 2

Diagrama de interaccion pag. 60

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Primer Historia de usuario entrega 5 -Diseño pantalla lista programas Información bloques
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	-Pantalla información por bloque -Diagrama de interacción -Caso de uso 2 -Acta 5 y 6 Plan de iteración entrega 4

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 7

15 de Octubre del 2017

Analista

-Aprobada primer historia de usuario entrega 5

Historia de Usuario – SOON UCC	
Dependencia	ANALISTA
Nombre de la Historia	LISTA PROGRAMAS
Descripción	Como usuario de la aplicación móvil SOON UCC deseo poder visualizar una lista de programas de pregrado ofertadas en la Universidad Cooperativa sede Bogotá donde pueda consultar información actualizada de todas y cada una de ellas.
Criterios de Aceptación	Lista con todos las carreras de pregrado ofertadas en Bogotá.
	En la lista debe ir el nombre de la carrera junto al logo de la Universidad Cooperativa de Colombia
	Al seleccionar una carrera debe aparecer la respectiva información actualizada.

Tabla 6 Historia de usuario 3 elaborado por Autores

Documentador y gestor de la información

-Aprobada información bloques, continuar realizando la investigación

Programador

-Aprobada pantalla información por bloque

Gerente y SQA

-Realizada Acta 5 y 6.Plan de iteración entrega 4.

-Plan de iteracion 4

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista	-Segunda Historia de usuario entrega 5
	Diseñador Grafico	-Diseño pantalla calculadora créditos
Hernandez Ruiz Jonathan Alexander	Tester	Información bloques
	Documentador y gestor de la información	-Mapa con ubicación por bloque
Hernandez Ruiz Jonathan Alexander	Programador	-Diagrama de secuencia
	Diseñador lógico	-Caso de uso 3
Hernandez Ruiz Jonathan Alexander	DBA	-Acta 7 y 8
	Gerente y SQA	Plan de iteración entrega 5

Tabla 7 Plan de iteracion 4 elaborado por Autores

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 8

22 de Octubre del 2017

Analista

-Aprobada segunda historia de usuario entrega 5

Historia de Usuario – SOON UCC	
Dependencia	ANALISTA
Nombre de la Historia	CALCULADORA CREDITOS
Descripción	Como usuario de la aplicación móvil SOON UCC deseo poder calcular el valor de la matrícula respecto al valor de créditos.
Criterios de Aceptación	Un campo en el cual pueda ingresar la cantidad de créditos teniendo en cuenta la información de la carrera.
	Un botón para que realice el cálculo y arroje un total

Tabla 8 Historia de usuario 4 elaborado por Autores

Diseñador grafico

-Aprobado diseño pantalla calculadora créditos



Imagen 5 Diseño 4 elaborado por Autores

Diseñador lógico

-Aprobado diagrama de secuencia y casos de uso 3.

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Segunda Historia de usuario entrega 5 -Diseño pantalla calculadora créditos Información bloques
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	-Mapa con ubicación por bloque -Diagrama de secuencia -Caso de uso 3 -Acta 7 y 8 Plan de iteración entrega 5

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 9

29 de Octubre del 2017

Documentador y gestor de la información

-Aprobada información bloques

Programador

-Aprobado mapa con ubicación por bloque

Gerente y SQA

-Realizada actas 7 y 8. Plan de iteración entrega 5

-Plan de iteracion 5

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Historia de usuario entrega 6 -Diseño pantalla llamadas -Plan de prueba -Información carreras
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	-Pantalla lista carreras -Pantalla calculadora créditos. -Modelo de datos -Modelo entidad relacion -Base de datos CarrerasUcc -Caso de uso 4 -Acta 9 y 10 Plan de iteración entrega 6

Tabla 9 Plan de iteracion 5 elaborado por Autores

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 10

05 de Noviembre del 2017

Analista

-Aprobada historia de usuario entrega 6

Historia de Usuario – SOON UCC	
Dependencia	ANALISTA
Nombre de la Historia	LLAMADAS DE EMERGENCIA
Descripción	Como usuario de la aplicación móvil SOON UCC deseo tener varias opciones para realizar llamadas en caso de emergencia.
Criterios de Aceptación	Una serie de botones que ejecuten la llamada a las respectivas entidades de emergencia.
	Se debe advertir al usuario antes de que se ejecute la llamada que la acción generara un costo en su operador de telefonía móvil.

Tabla 10 Historia de usuario 5 elaborado por Autores

Diseñador grafico

-Aprobado diseño pantalla llamadas



Imagen 6 Diseño 5 elaborado por Autores

Diseñador lógico

-Aprobado modelo de datos y modelo entidad relación

DBA

-Aprobada base de datos CarrerasUCC

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Historia de usuario entrega 6 -Diseño pantalla llamadas -Plan de prueba -Información carreras
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	-Pantalla lista carreras -Pantalla calculadora créditos. -Modelo de datos -Modelo entidad relación -Base de datos CarrerasUcc -Caso de uso 4 -Acta 9 y 10 Plan de iteración entrega 6

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 11

12 de Noviembre del 2017

Tester

-Aprobado plan de prueba

Documentador y gestor de la información

-Aprobada información carreras

Programador

-Aprobada pantalla lista carreras y pantalla calculadora créditos.

Diseñador lógico

-Aprobado caso de uso 4

Gerente y SQA

-Realizadas actas 9 y 10. Plan de iteración entrega 6

-Plan de iteracion 6

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Pruebas sobre la aplicación -Manual de usuario -Manual del sistema -Números de emergencia
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	-Pantalla llamadas de emergencia -Modelo vista -Patrones de desarrollo -Diccionario de datos -Caso de uso 5 -Acta 11 y 12

Tabla 11 Plan de iteracion 6 elaborado por Autores

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 12

19 de Noviembre del 2017

Documentador y gestor de la información

- Aprobado manual de usuario
- Números de emergencia

Diseñador lógico

- Aprobado modelo vista, patrones de desarrollo y diccionario de datos

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista	-Pruebas sobre la aplicación
	Diseñador Grafico	-Manual de usuario
Higuera Rojas Sergio Alejandro	Tester	-Manual del sistema
	Documentador y gestor de la información	-Números de emergencia
Hernandez Ruiz Jonathan Alexander	Programador	-Pantalla llamadas de emergencia
	Diseñador lógico	-Modelo vista
Hernandez Ruiz Jonathan Alexander	DBA	-Patrones de desarrollo
	Gerente y SQA	-Diccionario de datos
Hernandez Ruiz Jonathan Alexander		-Caso de uso 5
		-Acta 11 y 12
Hernandez Ruiz Jonathan Alexander		Plan de iteración entrega 7

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA 13

26 de Noviembre del 2017

Tester

-Continuar pruebas sobre la aplicación

Diseñador lógico

-Terminar manual del sistema

-Aprobado caso de uso 5

Programador

-Aprobada pantalla llamadas de emergencia

Gerente y SQA

-Realizadas actas 11 y 12 y plan de iteración entrega 7

-Plan de iteracion 7

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Pruebas sobre la aplicación -Manual del sistema
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	-Corregir y optimizar la aplicacion -Acta 13 y final

Tabla 12 Plan de iteracion 7 elaborado por Autores

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

-ACTA FINAL

03 de Diciembre del 2017

Resultados:

Después de haber realizado 13 reuniones en donde se desarrollaron 7 planes de iteración (Springs) se logró realizar una aplicación para dispositivos móviles con un módulo para ubicar los distintos bloques de la Universidad Cooperativa de Colombia, otro modulo para realizar el cálculo de los créditos respecto al valor definido por carrera y por ultimo un módulo para realizar llamadas de emergencia.

Por lo tanto queda constancia en esta acta final de que la aplicación móvil SOON cumple con el plan de desarrollo definido y se le han aplicado pruebas de desarrollo para su correcto funcionamiento. Se realiza manual de usuario y de sistema para su guía.

Miembro	Rol	Entregable
Higuera Rojas Sergio Alejandro	Analista Diseñador Grafico Tester Documentador y gestor de la información	-Pruebas sobre la aplicación -Manual del sistema
Hernandez Ruiz Jonathan Alexander	Programador Diseñador lógico DBA Gerente y SQA	-Corregir y optimizar la aplicación -Acta 13 y final

Control

Miembro	Estados
Higuera Rojas Sergio Alejandro	Asiste-Colabora
Hernandez Ruiz Jonathan Alexander	Asiste-Colabora

Pruebas de desarrollo

La funcion del tester fue analizar usar y vivir las primeras impresiones del uso de la aplicación móvil SOON, al tan solo ingresar a este aplicativos encontramos un menú muy interactivo y un título muy atractivo y muy representativo de las funciones.



Imagen 7 Pruebas 1 elaborado por Autores

Después de elegir una de las opciones como lo es (Bloques) encontramos un portal interactivo básico, pero a su vez muy útil y práctico donde podemos elegir la sede en la cual estamos interesados y cuenta con gran variedad de fotos por cada sede podemos encontrar además de mapas formas de llegar una breve descripción que nos muestra que podemos encontrar en estas sedes o que es lo más destacado.

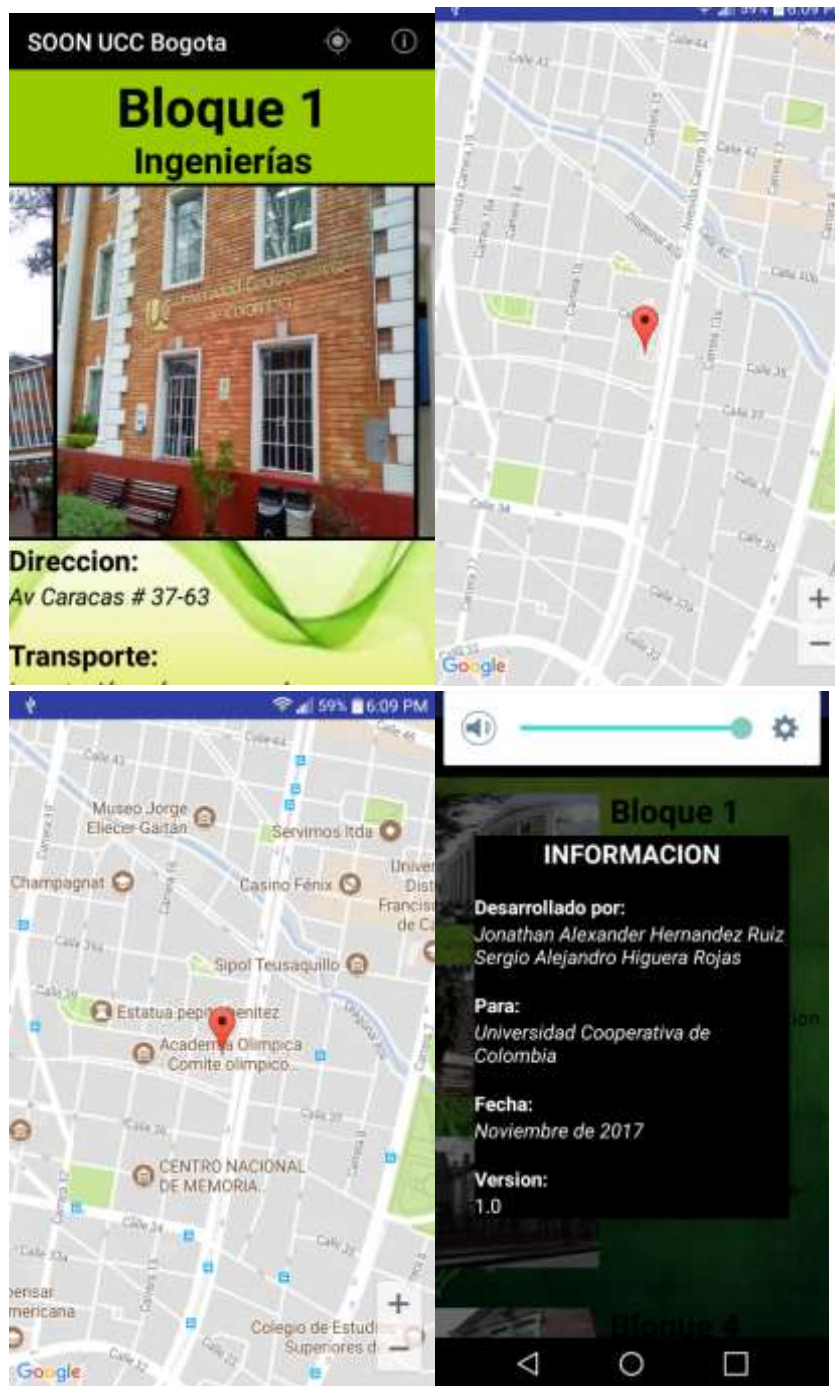


Imagen 8 Pruebas 2 elaborado por Autores

El módulo de emergencias es muy fácil de usar en caso de presentarse la necesidad de llamar a la celaduría o alguna ambulancia podremos hacerlo através de esta aplicación de forma ágil y segura solo con tan solo 2 clicks estaremos haciendo una llamada la cual puede salvar una vida porque cada segunda cuenta.



Imagen 9 Pruebas 3 elaborado por Autores

Esta aplicación afortunadamente también cuenta con una herramienta muy útil la cual nos permite calcular el valor de semestre por créditos, en esta aplicación podremos encontrar mucha información como el valor de cada carrera así como también las carreras que se imparten cantidad de créditos y lo mejor en una sola aplicación podremos calcular cuánto nos cuesta cada semestre para facilitarnos la vida a la hora de inscribir materia todas las funciones que encontramos en SOON nos facilitan la vida como estudiante o persona del común interesada en la Universidad.



Imagen 10 Pruebas 4 elaborado por Autores

3.2 PATRONES

-Patrones estructurales

Para el desarrollo de la aplicación se usaron los siguientes patrones estructurales:

Adapter: Permite a dos clases con diferentes interfaces trabajar entre ellas, a través de un objeto intermedio con el que se comunican e interactúan.

Activity: Permite a una clase utilizar los componentes necesarios para crear un controlador que se encarga de los layouts de la aplicación

SQLite: Funciona como una librería de Android

-Patrones creacionales

Para el desarrollo de la aplicación se usaron el siguiente patron creacional:

Builder: Separa la creación de un objeto complejo de su estructura, de tal forma que el mismo proceso de construcción nos puede servir para crear representaciones diferentes.

-Patrones de comportamiento

Command: Son objetos que encapsulan una acción y los parámetros que necesitan para ejecutarse.

3.3 CASOS DE USO

3.3.1. Diagrama de casos de uso

Diagrama de Casos de Uso

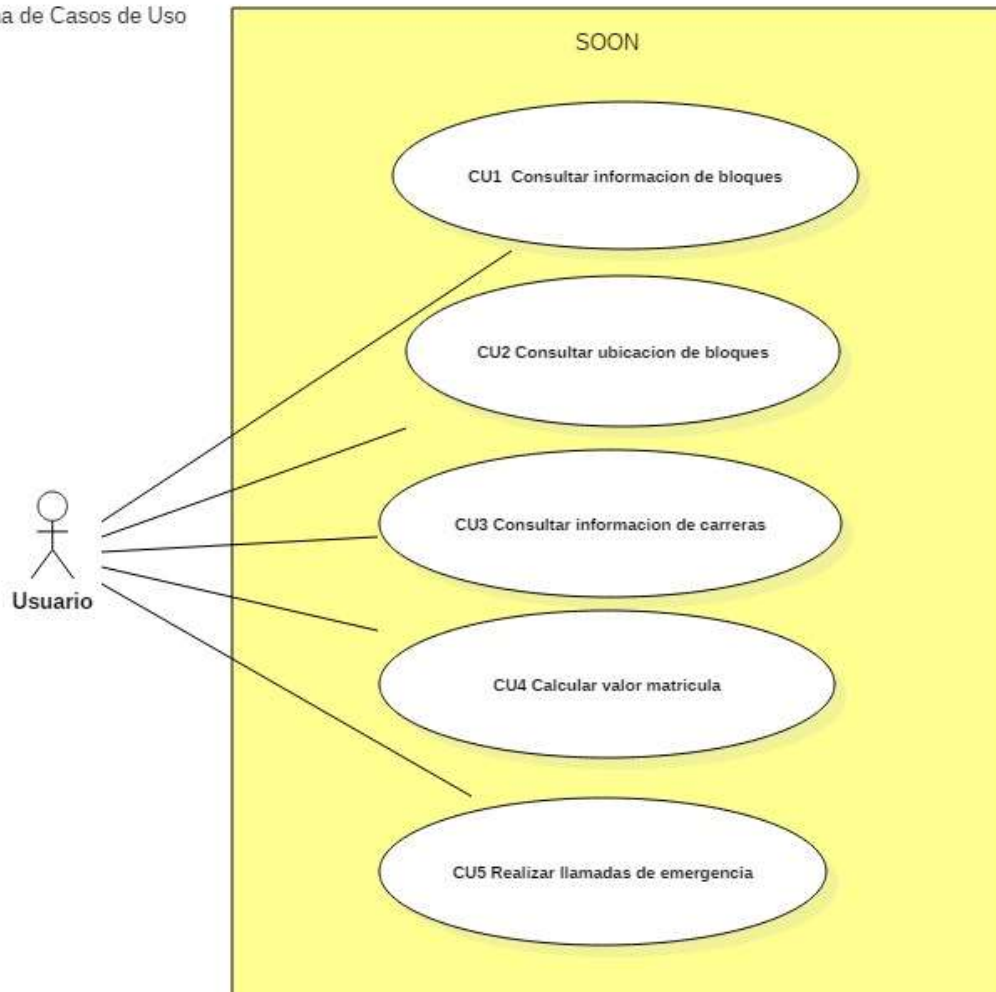


Grafico 6 Casos de uso elaborado por Autores

3.3.2. Caso de uso 1 – Consultar información bloques

CU-01	Consultar Información de bloques	
Versión	1.0 (17/09/2017)	
Actores	Usuario	
Referencias	De Requerimientos: Historia de Usuario – Lista bloques	
Objetivo	Consultar información estática de los diferentes bloques de la Universidad Cooperativa sede Bogotá.	
Precondición	El usuario ingresa mediante un botón de la aplicación a una lista de los bloques donde podrá seleccionar uno para obtener mayor información.	
Descripción	La aplicación deberá comportarse como se describe a continuación, cuando el usuario ingrese a la lista de bloques	
Secuencia Normal	Paso	Acción
	1	El usuario visualiza una lista de todos los bloques con una foto y una descripción
	2	El usuario selecciona uno de los bloques
	3	La aplicación muestra la información del bloque seleccionado.
	4	La información debe tener número del bloque, descripción, dirección, transporte, características e ubicación.
Postcondición	El usuario tendrá la opción de regresar a la lista para obtener información de otro bloque.	
Excepciones	Paso	Acción
Comentarios		

Tabla 13 Caso de uso 1 elaborado por Autores

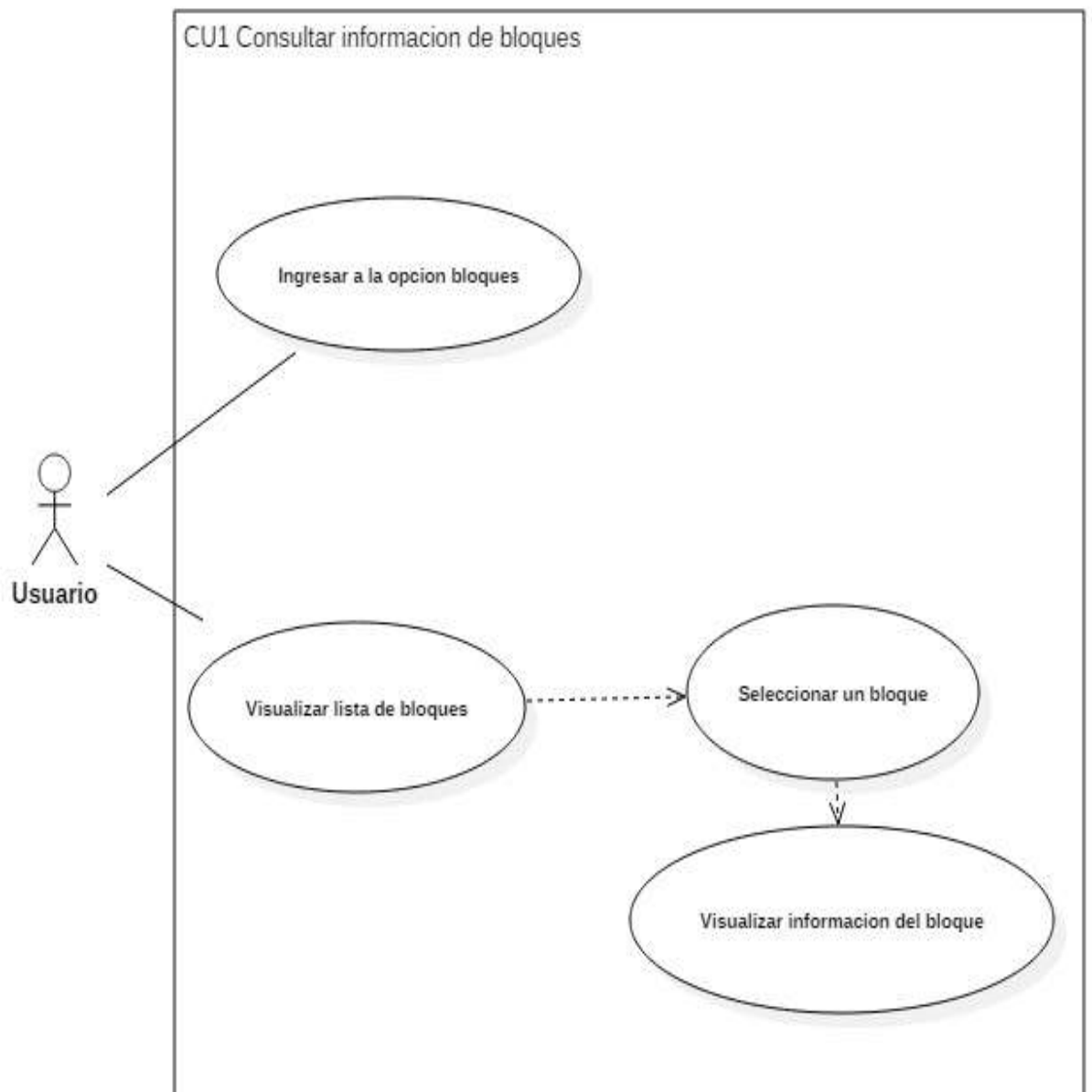


Grafico 7 Caso de uso 1 elaborado por Autores

3.3.3. Caso de uso 2 – Consultar Ubicación de bloques

CU-02	Consultar Ubicación de bloques	
Versión	1.0 (17/09/2017)	
Actores	Usuario	
Referencias	De Requerimientos: Historia de Usuario – Ubicación bloques	
Objetivo	Consultar la ubicación de los diferentes bloques de la Universidad Cooperativa sede Bogotá.	
Precondición	El usuario ingresa mediante un botón de la aplicación a un mapa con la ubicación de los bloques.	
Descripción	La aplicación deberá comportarse como se describe a continuación, cuando el usuario ingrese al mapa.	
Secuencia Normal	Paso	Acción
	1	El usuario visualiza un mapa con la ubicación del bloque seleccionado.
	2	Tiene la opción de cómo llegar si el usuario lo desea.
	3	El usuario puede manipular el mapa.
	4	En caso de utilizar la opción de cómo llegar debe ejecutar el Google Maps.
Postcondición	El usuario tendrá la opción de regresar a la información del bloque seleccionado.	
Excepciones	Paso	Acción
		<i>Cuando el usuario oprima la opción de Cómo llegar?</i>
	4	E1.Desde este punto la aplicación que ejecutara las funciones será Google Maps, por lo tanto el uso de datos móviles no es responsabilidad de SOON
Comentarios	Se sugiere crear un icono de ayuda que contenga instrucciones de cómo usar el mapa.	

Tabla 14 Caso de uso 2 elaborado por Autores

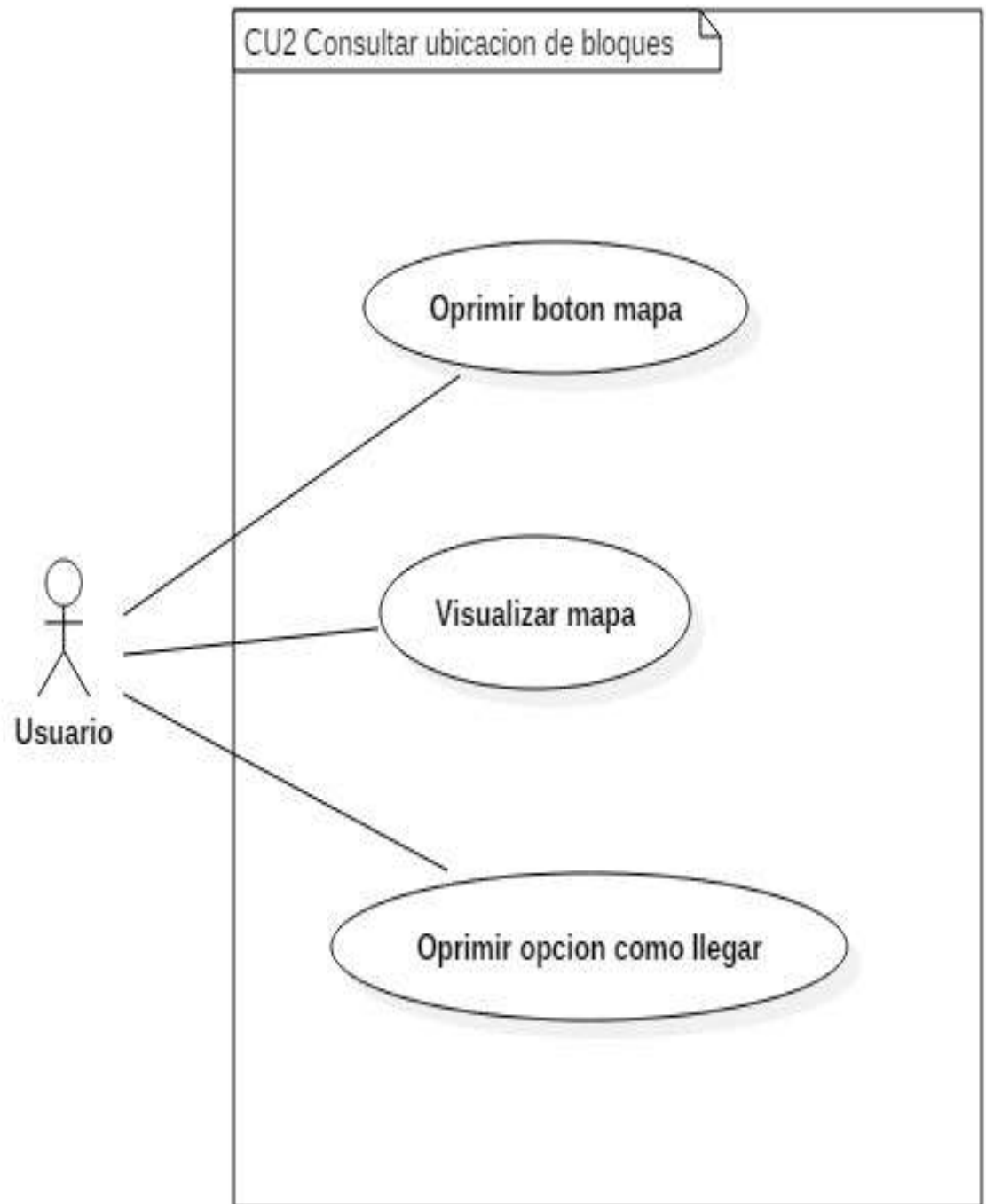


Grafico 8 Caso de uso 2 elaborado por Autores

3.3.4. Caso de uso 3 – Consultar informacion de programas

CU-03	Consultar Información de programas	
Versión	1.0 (17/09/2017)	
Actores	Usuario	
Referencias	De Requerimientos: Historia de Usuario – Lista programas	
Objetivo	Consultar información estática de las diferentes programas de pregrado ofertadas en la Universidad Cooperativa sede Bogotá.	
Precondición	El usuario ingresa mediante un botón de la aplicación a una lista de las carreras donde podrá seleccionar una para obtener mayor información.	
Descripción	La aplicación deberá comportarse como se describe a continuación, cuando el usuario ingrese a la lista de bloques	
Secuencia Normal	Paso	Acción
	1	El usuario visualiza una lista de todas las carreras con el logo de la Universidad.
	2	El usuario selecciona uno de los programas.
	3	La aplicación muestra la información de la carrera seleccionada.
	4	La información debe tener nombre de la carrera, ciudad, SNIES, semestres, créditos, valor del crédito.
Postcondición	El usuario tendrá la opción de regresar a la lista para obtener información de otra carrera.	
Excepciones	Paso	Acción
Comentarios		

Tabla 15 Caso de uso 3 elaborado por Autores

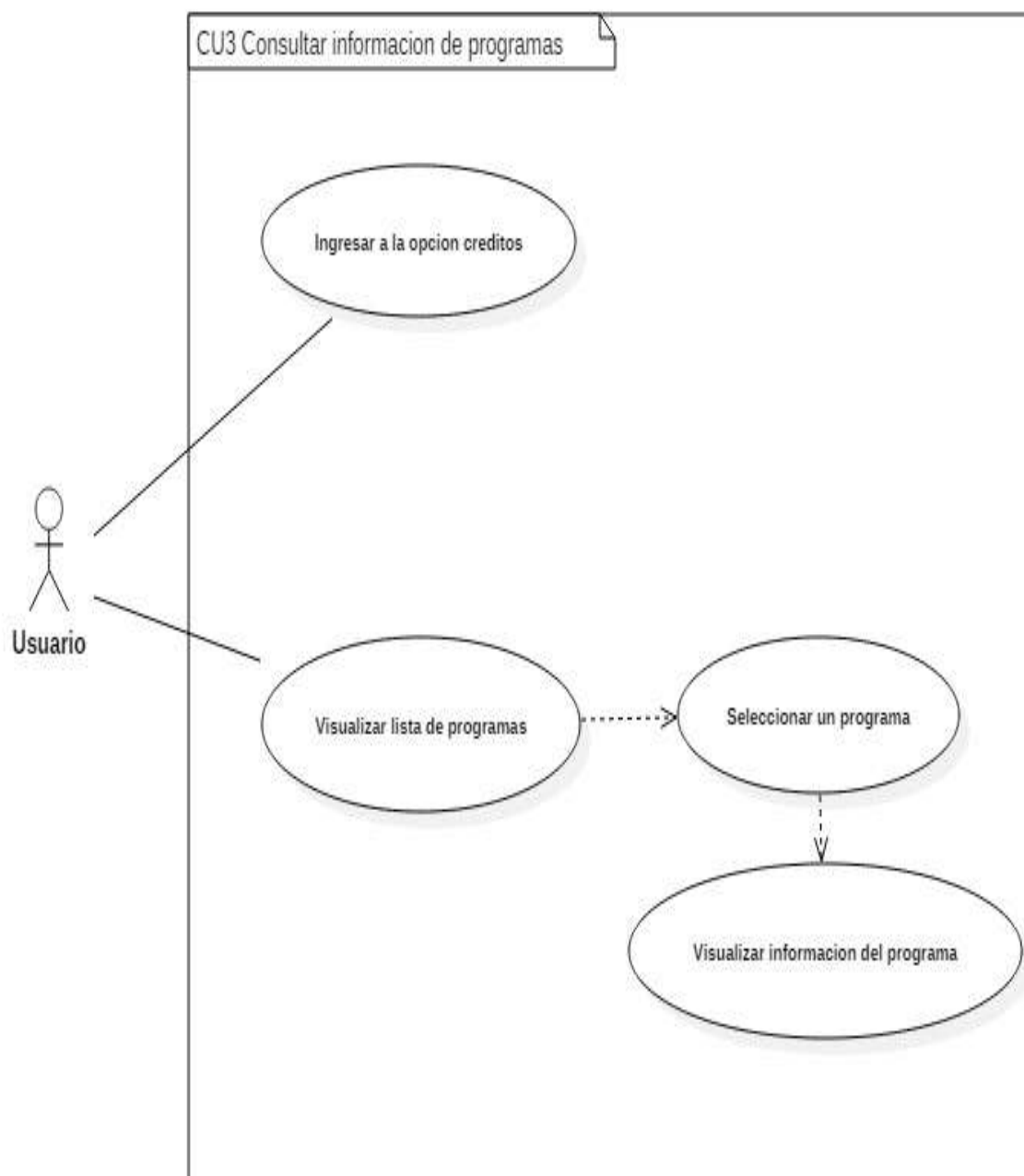


Grafico 9 Caso de uso 3 elaborado por Autores

3.3.5. Caso de uso 4 – Calcular el valor de la matricula

CU-04	Calcular valor de la matricula	
Versión	1.0 (17/09/2017)	
Actores	Usuario	
Referencias	De Requerimientos: Historia de Usuario – Calculadora de créditos	
Objetivo	Calcular el valor de la matricula respecto al valor de los créditos de la carrera solicitada.	
Precondición	El usuario ingresa la cantidad de créditos para calcular el valor de la matricula teniendo en cuenta el valor del crédito, la aplicación calcula el valor total .	
Descripción	La aplicación deberá comportarse como se describe a continuación, cuando el usuario oprima la opción calcular.	
Secuencia Normal	Paso	Acción
	1	El usuario visualiza el cálculo total del valor de la matricula respecto al valor de créditos ingresados
	2	El usuario tiene la opción de ingresar otra cantidad para calcular otro valor.
	3	
	4	
Postcondición	El usuario tendrá la opción de regresar a la lista para obtener el valor del crédito de otro programa.	
Excepciones	Paso	Acción
Comentarios		

Tabla 16 Caso de uso 4 elaborado por Autores

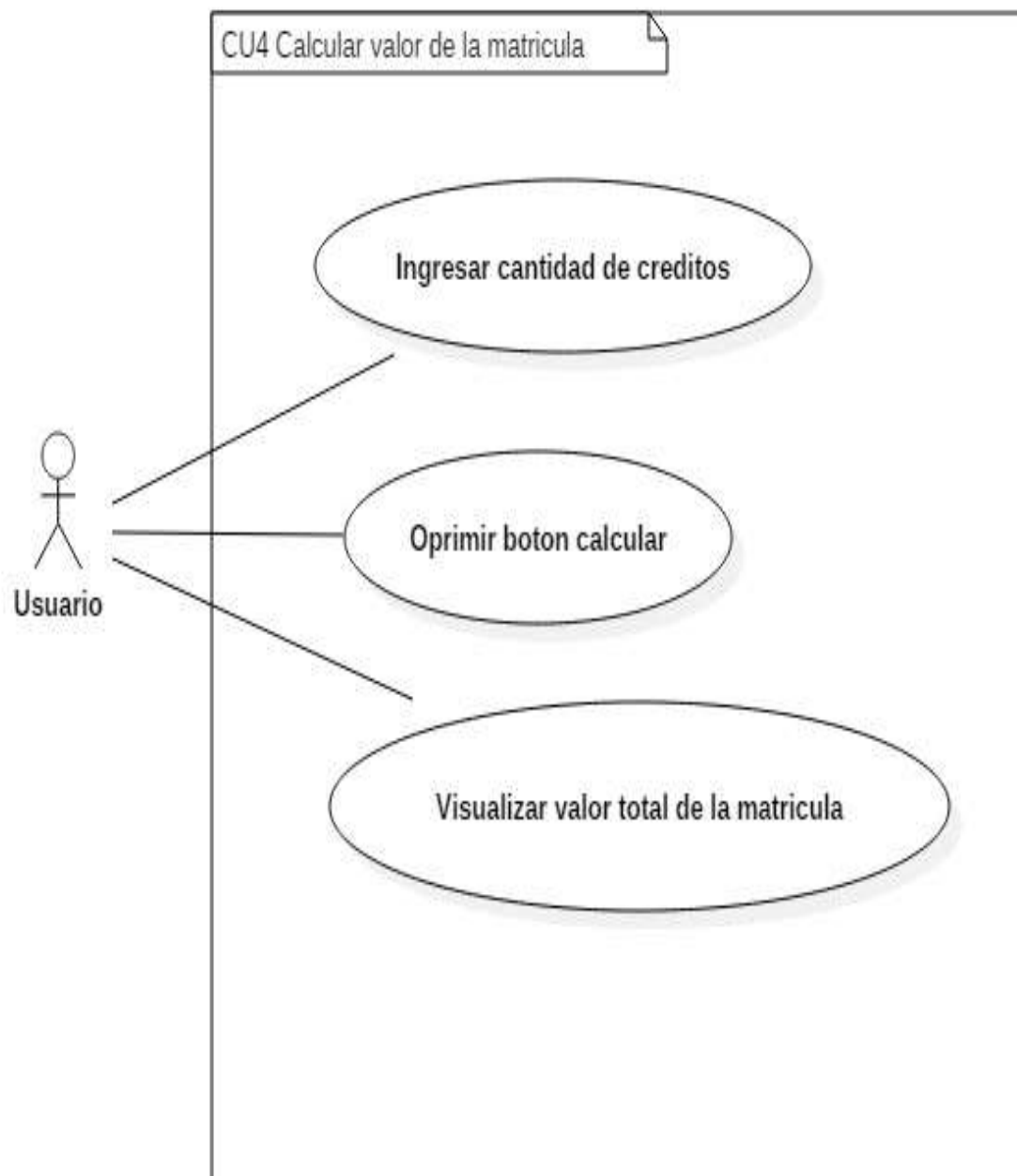


Grafico 10 Caso de uso 4 elaborado por Autores

3.3.6. Caso de uso 5 – Realizar llamadas de emergencia

CU-05	Realizar llamadas de emergencia	
Versión	1.0 (17/09/2017)	
Actores	Usuario	
Referencias	De Requerimientos: Historia de Usuario – Llamadas de emergencia	
Objetivo	Realizar llamadas mediante uno o más botones SOS en casos de emergencia.	
Precondición	El usuario ejecuta mediante un botón de la aplicación una llamada a una entidad de emergencia vinculada con la Universidad.	
Descripción	La aplicación deberá comportarse como se describe a continuación, cuando el usuario ingrese a la lista de bloques	
Secuencia Normal	Paso	Acción
	1	El usuario oprime el botón SOS y este le preguntara si desea realizar la llamada.
	2	Si se confirma la llamada debe ejecutarse inmediatamente
	3	Si no se confirma la llamada debe volver a la vista.
Postcondición	El usuario tendrá la opción de elegir entre hacer o no la llamada por costos en el operador de telefonía móvil.	
Excepciones	Paso	Acción
Comentarios		

Tabla 17 Caso de uso 5 elaborado por Autores

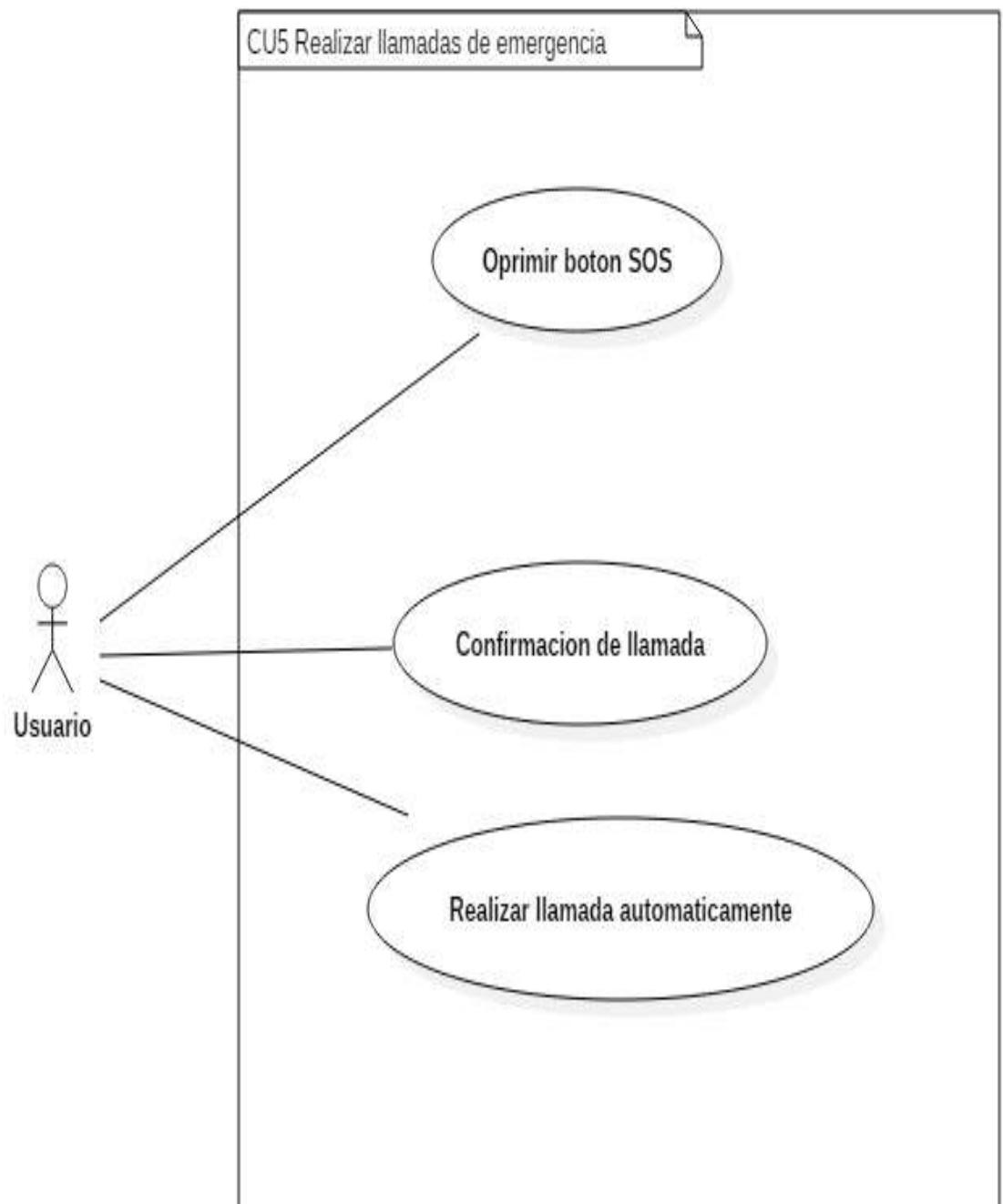
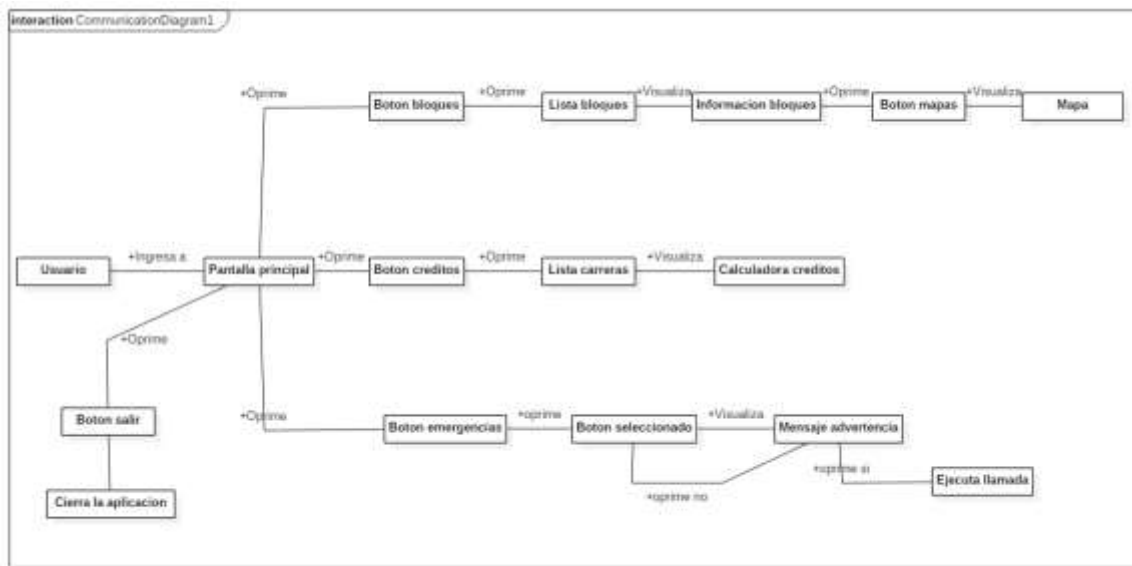


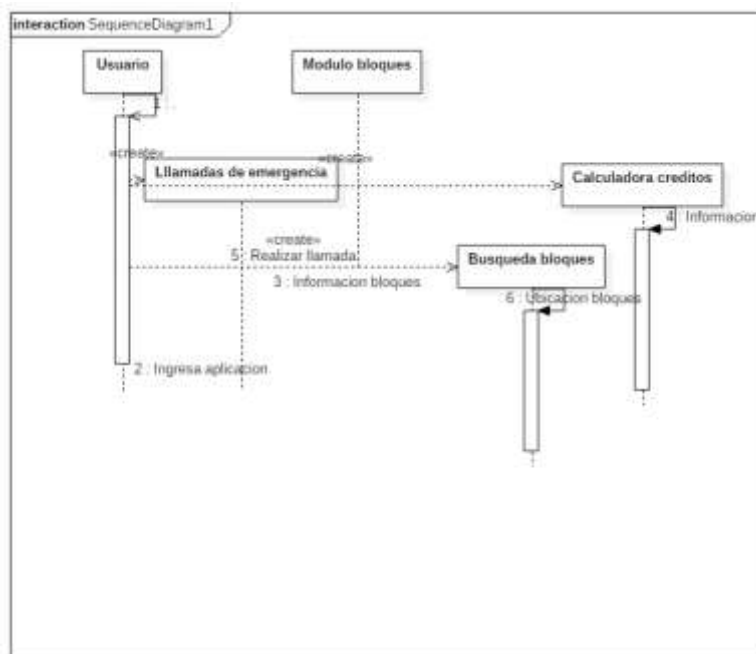
Grafico 11 Caso de uso 5 elaborado por Autores

3.6 DIAGRAMA DE INTERACCIÓN



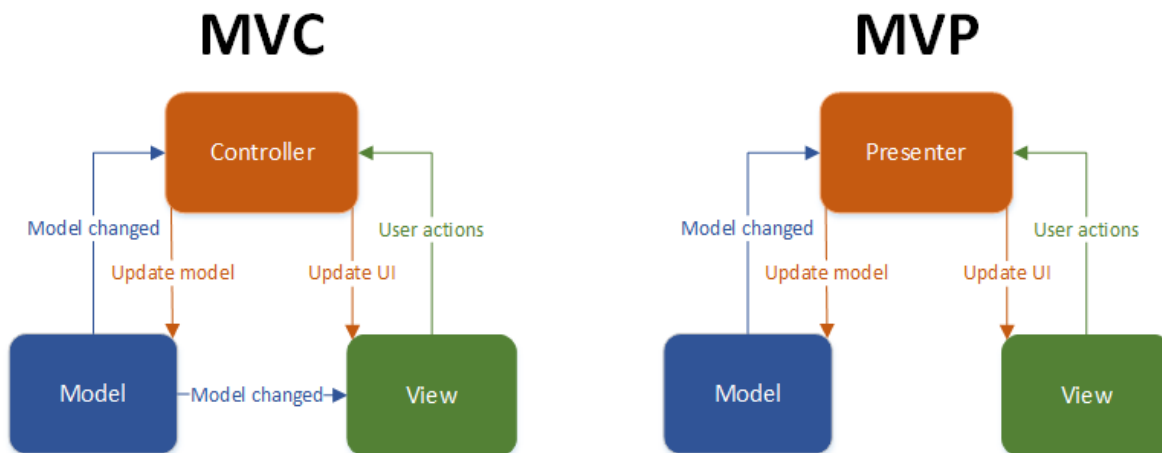
Grafica 14 Diagrama de interaccion elaborado por Autores

3.7 DIAGRAMA DE SECUENCIA

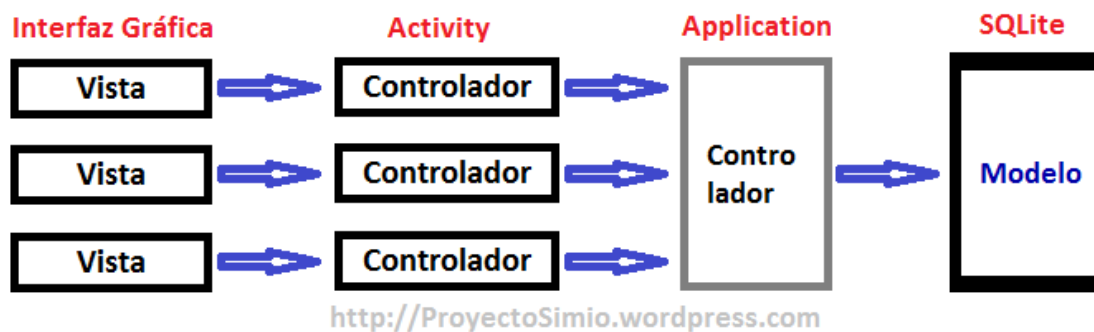


Grafica 15 Diagrama de secuencia elaborado por Autores

3.8 MODELO VISTA



Grafica 16 MVC1 elaborado por <https://www.techyourchance.com/mvp-mvc-android-1/>



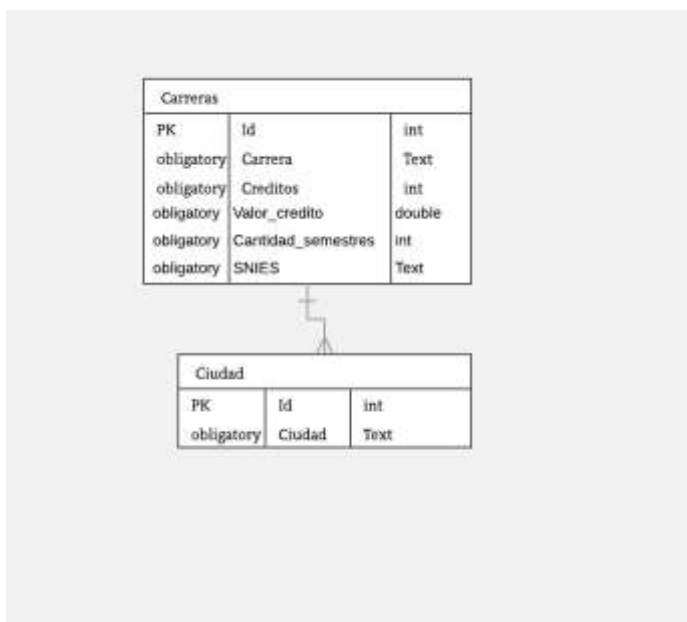
Grafica 17 MVC2 <http://www.proyectosimio.com/es/programacion-android-base-de-datos-i-modelo-vista-controlador/>

3.9 MODELO DE DATOS

Table	Carreras				
Name	Data Type	Pk	Fk	Nullable	Description
Id	int	True			Identificador único de carrera
Carrera	Text				Nombre de la carrera
Creditos	int				Cantidad de creditos de la carrera
Valor_credito	double				Valor del credito de la carrera
Cantidad_semestres	int				Cantidad de semestres de la carrera
Ciudad	Text				Ciudad donde se oferta la carrera
SNIES	Text				Codigo SNIES de la carrera

Tabla 18 Modelo de datos elaborado por Autores

3.10 MODELO RELACIONAL



Grafica 18 Modelo relacional elaborado por Autores

3.11 DICCIONARIO DE DATOS

Nombre de Archivo: CarrerasUCC			Fecha de creacion:05/11/2017
Descripción: Base de datos que contendrá la información de las carreras de pregrado de la UCC			
Campo	Tamaño	Tipo de dato	Descripción
Id	30	int	Clave de carrera
Carrera	50	text	Nombre de la carrera
Créditos	3	int	Cantidad de créditos de la carrera
Valor_credito	30	double	Valor del crédito de la carrera
Cantidad_semestres	30	int	Duración aproximada de la carrera
Ciudad	30	text	Ciudad donde se ofrece la carrera
SNIES	30	int	Código SNIES de la carrera

Tabla 19 Diccionario de datos elaborado por Autores

4. CONCLUSIONES Y RECOMENDACIONES

-Conclusiones

Es un proyecto muy atractivo e interesante que puso a prueba y en práctica todos los conocimientos adquiridos en la universidad a lo largo de la carrera, también nos hizo esforzarnos un poco más e ir algo más lejos investigar más herramientas lenguajes y formas de llevar acabo nuestra idea y lograrlo.

SOON UCC es una aplicación desarrollada en Android studio para dispositivos móviles la cual está enfocada para facilitar la información de los bloques de la universidad cooperativa de Colombia por medio de ubicación satelital, pensada en los estudiantes administrativos y externos que estén interesados en la UCC. ¿Por qué es importante esta aplicación? SOON nos permitirá solucionar la desinformación a la hora de movilizarnos a las distintas sedes de la UCC puesto que cuenta con más de 28 bloques en Bogotá, sin la aplicación podemos notar total desinformación puesto que realizamos el proceso de inscripción una materia y el sistema solo nos arroja el bloque en el cual se dará la clase sin la dirección ni cómo llegar las sedes de la UCC están distribuidas en los distintas zonas de Bogotá esto causa inconformismo en los estudiantes este mismo problema está presente en con los egresados profesores administrativos y a su vez con quienes quieren formar parte de la UCC, en base a esto esta aplicación no solo nos facilitara la forma de llegar al universidad sin que también cuenta con un módulo de valor crédito el cual está enfocado a informarle a todos los que estudian una carrera en a UCC o quieran estudiar en la misma el valor de cada carrera la cantidad de créditos así como también la duración esta calculadora de créditos facilitara el cálculo a la hora de iniciar nuestros estudios, y ya por ultimo pero no menos importante tiene incluida una función de emergencia en la cual encontramos las distintas opciones de marcación (celaduría, Emermedica, y personal calificado) a través de un solo click en caso de necesitarlo.

-Recomendaciones

1. La aplicación debe darse a conocer entre los distintos estudiantes e interesados lo cual nos lleva a que debemos incluirla en un motor de descargas como lo es la APP STORE
2. al implementar esta aplicación y obtener grana acogida en los estudiantes será necesaria actualizarla de forma periódica lo cual nos hará implementar tiempo y recursos para mantener un servidor que envíe estas actualizaciones de forma esporádica.
3. Enfocándonos en la falencia de Bogotá dejamos de lado las demás seccionales de la universidad cooperativa de Colombia.
4. Aunque este enfocada a jóvenes que utilizan mucho las herramientas 2.0 en su vida social, no debemos dar por supuesto que conocen su funcionamiento.
5. Conforme la aplicación se expanda en información requerida por los estudiantes como foros videos y otras funciones esta aumentara el peso de la aplicación
6. Por el momento esta aplicación está enfocado a los celulares con sistema operativo Android lo cual ocasionara que quienes cuenten con otro sistema operativa la quieran descargar no lo puedan hacer por compatibilidad
7. Al implementar esta aplicación en celulares y Tablet con sistema operativo Android nos conllevara a que los interesados en la aplicación también busquen otras formas de acceder como un portal de internet interactivo que funcione de la forma que lo hace la aplicación.

5. REFERENCIAS BIBLIOGRÁFICAS

Robert C. Martin (2012). Código Limpio-Manual de estilo para el desarrollo ágil de software. USA: ANAYA.

Universidad Cooperativa de Colombia (2017). Página de inicio. Bogotá, Colombia: Recuperado de: <http://www.ucc.edu.co/Paginas/inicio.aspx>.(2017/10/15).

API Android Developers (2017).Bogotá, Colombia Recuperado de: <https://developer.android.com/reference/packages.html>(2017/11/15)

API Java (2017).Bogotá, Colombia Recuperado de: <https://developer.android.com/reference/packages.html>(2017/11/15)

Thierry Groussard (2015).JAVA 7 Los fundamentos del lenguaje Java. USA: ENI.

Jonathan S. Harbour (2015). JAVA SE 6 Game programming, third edition USA: Course technology.

Martin Fowler con Kendall Scott (1999).UML Gota a Gota, USA: PEARSON Prentice Hall

Craig Larman (2002).UML y Patrones, segunda edición USA: PEARSON Prentice Hall

Shari Lawrence (2002). Ingeniería de software Teoría y práctica, USA: PEARSON Prentice Hall

Ian Sommerville (2009). Ingenieria del software, USA: PEARSON Prentice Hall

Scrum.org(2017) Estados Unidos Recuperado de:

<https://www.scrum.org/> (2017/11/20)

Mobile Marketing Association Spain (2017) España Recuperado de:

<https://mmaspain.com/wp-content/uploads/2015/12/Libro-Blanco-Webs-Moviles.pdf>
(2017/11/20)

Wei-Meng Lee Traducido por Javier Díaz Domés (2013) desarrollo de aplicaciones ganadoras Editor Anaya Multimedia

Mario Zechner (2011) Desarrollo de juegos para Android Editor Anaya Multimedia

Sébastien Pérochon (2012) Guía de desarrollo de aplicaciones para Smartphones y Tabletas Editor ENI EDICIONES

Joan Ribas Lequerica (2011) Desarrollo de aplicaciones para Android Manuales Imprescindibles Editor Anaya Multimedia

Jesús Tomás Gironés (2014) El gran libro de Android Editorial Marcombo

Maximiliano paredes Velasco (2014) Programación multimedia y dispositivos móviles (grado superior) Editorial Ra-ma s.a. editorial y publicaciones

Nolasco Jorgedelau (2016) Android desarrollo de aplicaciones móviles Editorial Delau