UCAB Universidad Católica ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

Aplicativo móvil para consultar y compartir prendas de vestir mediante la utilización de estrategias de comparación por tallas

Este Jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado: Dieciseis (16) Puntos

JURADO EXAMINADOR

Firma:	Firma:	Firma:
Nombre: Valeria León	Nombre: FANNY HERNANDEZ	Nombre: RAFAEL LARA CAMPOS

REALIZADO POR	**Luis Arturo Rodríguez Pacheco**
TUTOR EMPRESARIAL	**Carlo Magurno**
TUTOR ACADÉMICO	**Fanny Hernández**
FECHA	**Junio 2017**

UCAB Universidad Católica ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

Aplicativo móvil para consultar y compartir prendas de vestir mediante la utilización de estrategias de comparación por tallas

TRABAJO INSTRUMENTAL DE GRADO

Presentado ante la

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

Como parte de los requisitos para optar al título de

INGENIERO EN INFORMÁTICA

REALIZADO POR	**Luis Arturo Rodríguez Pacheco**
TUTOR EMPRESARIAL	**Carlo Magurno**
TUTOR ACADÉMICO	**Fanny Hernández**
FECHA	**Junio 2017**

DEDICATORIA

A mis padres Lilia Josefina Pacheco Rodríguez y

Héctor José Rodríguez Díaz, quienes forman

parte esencial de mi vida y de este logro,

guiándome en cada etapa de mi educación.

A mi abuela Julia Josefina Díaz Cabrera,

quien me ha acompañado en cada etapa

de mi vida apoyándome y aconsejándome.

¡Este logro es para ustedes!

Luis Arturo Rodríguez Pacheco

AGRADECIMIENTOS

Agradezco a Dios por guiarme siempre por el buen camino y permitirme cumplir esta meta.

Agradezco a mi familia, por acompañarme en cada etapa de mi desarrollo como profesional. Especialmente a Ana Teresa Díaz, Reina Díaz, Aimara Ávila, Maritza Pacheco e Irma Pacheco por su constante fe en mí y apoyo incondicional en cada etapa de mi educación.

Agradezco a mis hermanos Rafael Rodríguez, Rodolfo Alarcón, Arturo Ávila y Héctor Rodríguez por su constante apoyo, por acompañarme y aconsejarme tanto en el ámbito universitario como en la vida.

Agradezco a mis amigos y compañeros de estudio Ebrain Flores, Daniel Moncada, Daniel Rendón, Gerardo Astone, Joselyn González, Michelle Serrano, Gabriela Grillo, Valentina Espinal y Bárbara Rodríguez. A mis amigos y compañeros de trabajo Mimia Lo, Marco Franceschi y Alexander Díaz. Con quienes compartí durante toda el transcurso de la carrera alegrías y tristezas, reponiéndonos de los tropiezos y nunca bajando la cabeza.

Luis Arturo Rodríguez Pacheco

ÍNDICE

Dedicatoria i

Agradecimientos ii

Índice iii

Índice de Figuras viii

Índice de Tablas x

Sinopsis xi

Capítulo I – El Problema 1

I.1 Necesidades de la Empresa 1

I.2 Solución Propuesta 2

I.3 Objetivo General 3

I.4 Objetivos Específicos 4

I.4.1 Aporte Funcional 4

I.4.2 Aporte Tecnológico 4

I.5 Alcance 5

I.6 Limitaciones 10

I.7 Justificación 11

Capítulo II – Marco Teórico 12

II.1 Sistemas operativos 12

II.1.1 Sistemas Operativos Móviles 12

II.1.1.1 Sistema Operativo Android 12

II.1.1.2 Sistema Operativo iOS 13

II.2 Aplicaciones móviles 14

II.2.1 Aplicaciones Nativas 14

II.2.2 Aplicaciones *Web* 14

II.2.3 Aplicaciones Híbridas 15

II.3 Lenguajes de Programación 15

II.3.1 JavaScript 16

II.3.2 Java 16

II.4 Frameworks de desarrollo móvil 16

II.4.1 Apache Cordova 16

II.4.2 jQuery Mobile 17

II.4.3 Xamarin 17

II.5 Bases de Datos 17

II.5.1 Bases de datos relacionales 17

II.5.1.1 Gestores de bases de datos relacionales más usados 18

II.5.2 Bases de datos NoSQL 18

II.5.2.1 Clave – Valor 18

II.5.2.2 Orientadas a documentos 19

II.5.2.3 Orientadas a grafos 19

II.6 Servicios Web 19

II.6.1 Servicios web RESTful 19

II.7 Backend as a Service (BaaS) 21

II.8 Plataformas de *Cloud Computing* 22

II.8.1 Google Firebase 22

II.8.2 Amazon Web Services (AWS) 22

II.8.2.1 Amazon DynamoDB 23

II.8.2.2 Amazon SNS 23

II.9 Web crawler y Web scraping 24

Capítulo III – Marco Metodológico 25

III.1 Fases 25

III.2 Artefactos 26

III.3 Roles 27

III.4 Objetivos y sprints 27

Capítulo IV - Desarrollo 29

IV.1 Introducción 29

IV.2 Arquitectura del sistema 31

IV.3 Selección de tecnologías para el desarrollo del sistema 32

IV.3.1 Selección de tecnología para el desarrollo del aplicativo móvil 32

IV.3.2 Selección de tecnología para el almacenamiento de datos, notificaciones *push* y gestión de usuarios 34

IV.3.3 Selección de tecnologías para el desarrollo del motor de búsqueda 36

IV.4 Sprints 38

IV.4.1 Sprint 1 – Base de datos, Módulo de Registro y Módulo de Configuración inicial 39

IV.4.1.1 Módulo de Base de datos 39

IV.4.1.2 Desarrollo del módulo de Registro 41

IV.4.1.3 Desarrollo del módulo de Configuración inicial 42

IV.4.2 Sprint 2 – Módulo de Gestión de tallas y prendas 42

IV.4.2.1 Investigación de prendas, tallas y medidas de partes del cuerpo 43

IV.4.2.2 Gestión de prendas de vestir 44

IV.4.2.3 Gestión de tallas 45

IV.4.3 Sprint 3 – Módulo de Gestión de marcas y Módulo de Gestión de perfiles 45
IV.4.3.1 Gestión de marcas 46
IV.4.3.2 Gestión de perfiles 46
IV.4.4 Sprint 4 – Módulo de Gestión de contactos y Gestión de notificaciones 47
IV.4.4.1 Gestión de contactos 47
IV.4.4.2 Gestión de notificaciones 48
IV.4.5 Sprint 5 – Módulo de búsqueda, comparación y emparejamiento 49
IV.4.5.1 Recolección y almacenamiento de prendas de vestir 50
IV.4.5.2 Consulta prendas de vestir a través del aplicativo móvil 51
IV.4.6 Sprint 6 – Módulo de Configuraciones y ajustes 51
Capítulo V – Resultados 53
Capítulo VI - Conclusiones y Recomendaciones 60
VI.1 Conclusión 60
VI.2 Recomendaciones 62
Referencias Bibliográficas 64
ApénDiCe 68
Apéndice A – Artefactos de la metodología 68
Product Backlog 68
Sprint Backlog 70
Apéndice B – Artefactos del desarrollo 74
Sprint 1 – BD, Módulo de Registro y Módulo de Configuración inicial 74
Sprint 2 – Módulo de Gestión de tallas y prendas 77
Sprint 3 – Módulo de Gestión de marcas y Módulo de Gestión de perfiles 82

Sprint 4 – Módulo de Gestión de contactos y Gestión de notificaciones..... 82

Sprint 5 – Módulo de búsqueda, comparación y emparejamiento..... 84

Sprint 6 – Módulo de Configuraciones y ajustes 85

ÍNDICE DE FIGURAS

Figura 1. – Mercado mundial de SO de *smartphones.* Fuente: Smartphone OS Market Share, 2016 Q3 (2016). 14

Figura 2. – Fases Scrum. Fuente: *Scrum in Action - Cengage Learning* (2012). 26

Figura 3. – Mapa de objetivos y sprints [Elaboración propia] 28

Figura 4. – Flujo final del sistema [Elaboración propia] 30

Figura 5. – Arquitectura del sistema [Elaboración propia] 32

Figura 6. – Modelo de datos NoSQL (users) [Elaboración propia] 40

Figura 7. – Modelo de base de datos NoSQL 1 [Elaboración propia] 74

Figura 8. – Modelo de base de datos NoSQL 2 [Elaboración propia] 75

Figura 9. – Proceso de registro y configuración inicial [Elaboración propia] 76

Figura 10. – Tabla config de la base de datos local [Elaboración propia] 76

Figura 11. – Proceso de adición de prenda favorita [Elaboración propia] 77

Figura 12. – Proceso de modificación de prenda favorita [Elaboración propia] 77

Figura 13. – Proceso de adición de medida de parte del cuerpo [Elaboración propia] 78

Figura 14. – Proceso de modificación de medida de parte del cuerpo [Elaboración propia] 78

Figura 15. – Proceso de visualización y selección de marcas favoritas [Elaboración propia] 82

Figura 16. – Proceso de solicitud de conexión con contacto [Elaboración propia] 82

Figura 17. – Solicitud de medida de parte del cuerpo [Elaboración propia] 83

Figura 18. – Visualización de notificaciones [Elaboración propia] 83

Figura 19 – Modelo de datos MySql [Elaboración propia] 84

Figura 20 – Consulta de prendas [Elaboración propia] 85

Figura 21. – Configuraciones en el aplicativo móvil [Elaboración propia]..... 85

ÍNDICE DE TABLAS

Tabla 1. – Objetivos desarrollados por iteración [Elaboración propia] 30

Tabla 2. – Comparación entre Frameworks de desarrollo híbrido [Elaboración propia] 33

Tabla 3. – Comparación entre plataformas de *cloud computing* [Elaboración propia] 35

Tabla 4. – Comparación entre sistemas de gestión de bases de datos [Elaboración propia] 38

Tabla 5. – Prendas de vestir [Elaboración propia] 43

Tabla 6. – Tipos de prenda de vestir [Elaboración propia] 43

Tabla 7. – Product Backlog [Elaboración propia] 68

Tabla 8. – Sprint Backlog 001. [Elaboración propia] 70

Tabla 9. – Sprint Backlog 002. [Elaboración propia] 70

Tabla 10. – Sprint Backlog 003. [Elaboración propia] 71

Tabla 11. – Sprint Backlog 004. [Elaboración propia] 71

Tabla 12. – Sprint Backlog 005. [Elaboración propia] 71

Tabla 13. – Sprint Backlog 006. [Elaboración propia] 72

Tabla 14. – Sprint Backlog 007. [Elaboración propia] 72

Tabla 15. – Sprint Backlog 008. [Elaboración propia] 72

Tabla 16. – Sprint Backlog 009. [Elaboración propia] 73

Tabla 17. – Tallas y medidas (Hombre) [Elaboración propia] 79

Tabla 18. – Tallas y medidas (Mujer) [Elaboración propia] 80

Tabla 19. – Prendas y partes del cuerpo [Elaboración propia] 81

SINOPSIS

El presente Trabajo Instrumental de Grado (TIG) titulado "Aplicativo móvil para consultar y compartir prendas de vestir mediante la utilización de estrategias de comparación por tallas", está basado en un aplicativo móvil que tiene como principal funcionalidad la consulta de prendas de vestir que más se ajustan a las características físicas (medidas específicas del cuerpo) que posee el usuario que lo maneja además de otros aspectos establecidos por él, permitiendo que la búsqueda de prendas de vestir sea más exacta y acorde a las necesidades del usuario.

Para llevar a cabo el proyecto fue planteado el uso del marco de trabajo Scrum, utilizado para la gestión y desarrollo de software, basado en un proceso iterativo e incremental utilizado comúnmente en entornos de desarrollo ágil de *software*.

El desarrollo de este Trabajo Instrumental de Grado fue dividido en seis iteraciones (*sprints*) que facilitaron y agilizaron su elaboración, modulados de la siguiente manera: el *sprint* uno contempla el diseño e implementación del modelo de base de datos, el desarrollo del registro usuarios y la configuración inicial del aplicativo móvil; el *sprint* dos abarca la gestión de tallas y prendas de los usuarios; el *sprint* tres engloba la gestión de marcas y gestión de perfiles de usuarios; el *sprint* cuatro incluye la gestión de contactos y la gestión de notificaciones; el *sprint* cinco se encarga del desarrollo de la búsqueda, comparación y emparejamiento de las prendas de vestir; y, por último, el *sprint* seis que incluye el desarrollo de las configuraciones y ajustes del aplicativo móvil.

El diseño de los nuevos procesos para la búsqueda de prendas de vestir permitirá a los usuarios del aplicativo móvil obtener de una forma eficaz las prendas de vestir consultadas según las características físicas deseadas; permitiendo además, almacenar las prendas que se obtuvieron como resultado de la ejecución de éstos procesos y optar por la compra de las prendas de vestir directamente en el sitio web de Amazon.com.

CAPÍTULO I – EL PROBLEMA

I.1 NECESIDADES DE LA EMPRESA

Trascend Software Evolution C.A. es una empresa dedicada al diseño, desarrollo y mantenimiento de software de alta calidad, abarcando tanto el mercado nacional como internacional. Provee servicios durante todo el ciclo de desarrollo de *software* con foco en áreas de desarrollo *web*, móvil y de escritorio. Han desarrollado un portafolio en materia de desarrollo de aplicativos, que cuenta con servicios de alta calidad y desempeño, basándose en las mejores prácticas, estándares y patrones de diseño. Esta empresa se ha mantenido en constante crecimiento, buscando mejorar y desarrollar nuevos aplicativos.

Actualmente en el mercado, no existe un medio de conocimiento compartido de información, mediante el cual se realice búsquedas de prendas de vestir empleando mecanismos de comparaciones y emparejamientos con el uso de características propias de este tipo de productos, tales como tallas que hayan sido registradas por algún usuario y que a su vez permita compartir dichos productos con otros usuarios registrados.

Trascend Software Evolution se ha planteado como proyecto, el desarrollo de un aplicativo móvil dirigido a Tecnófilos[1], que permita a sus usuarios la búsqueda de prendas de vestir en un sitio *web* proveedor de este tipo de productos llamado **Amazon.com**, donde la principal característica o patrón de búsqueda sea las tallas de las prendas de vestir establecidas por el usuario.

En el aplicativo móvil, las personas podrán registrarse y crear o modificar sus perfiles, además podrán buscar y enviar solicitudes de amistad a otros usuarios registrados, a través de la agenda de contactos personal del dispositivo móvil del usuario.

[1] **Tecnófilos**: Personas con cierta afición a la tecnología o dispositivos relacionados con ella.

Con el uso del aplicativo móvil, el usuario podrá almacenar en su perfil sus tallas correspondientes a cada tipo de prenda que desee, medidas de cada parte del cuerpo (brazo, busto, pecho, dedos, pies, cabeza, cadera, pierna, cuello, hombro, muslo y cintura) y sus tipos de prendas preferidas, para luego publicarlas y que sus contactos puedan acceder a dicha información. Se desea que en el aplicativo móvil, luego de que un usuario haya registrado tallas y medidas de partes del cuerpo, éste pueda solicitar una búsqueda de prendas de vestir relacionadas a los datos ya registrados, para que a través de la interacción entre el aplicativo y **Amazon.com** se pueda establecer un emparejamiento entre las características de las prendas agregadas y las registradas en el sitio web, y de esta manera obtener como resultado las prendas de vestir alojadas o publicadas en el sitio web que más se ajustan a las especificaciones registradas en el aplicativo móvil por el usuario.

I.2 SOLUCIÓN PROPUESTA

En vista de la problemática establecida anteriormente, se plantea el desarrollo de un aplicativo móvil híbrido[2], que trabaje con un *framework*[3] de desarrollo multiplataforma y un sistema de gestión de base de datos compatible con dispositivos móviles, permitiendo que el aplicativo móvil se pueda ejecutar en distintos sistemas operativos como *iOS* y *Android*. Además se hará uso de una tecnología que permita realizar vinculaciones con *Cloud Storage*[4] y sincronización con datos en tiempo real.

El aplicativo móvil tendrá un módulo de registro, en donde el usuario tendrá la opción de registrarse mediante el ingreso de su número telefónico o a través de autenticación con cuentas en **Facebook**. Además se tendrá un módulo dedicado a la configuración inicial, donde se podrá ingresar el género del usuario y la talla de una

---

[2] **Aplicativo móvil hibrido**: Son aplicaciones móviles que están desarrolladas en lenguajes HTML, Javascript y CSS, y que permiten acceder a gran parte de las características del hardware del dispositivo.
[3] **Framework**: Esquema para el desarrollo y/o la implementación de una aplicación.
[4] **Cloud Storage (Almacenamiento en la nube)**: es un modelo de almacenamiento de datos basado en redes de computadoras donde los datos están alojados en espacios de almacenamiento virtualizados.

prenda en específico elegida por el usuario de la lista de prendas de vestir del aplicativo móvil.

El aplicativo móvil debe tener un módulo de creación y modificación de perfil, donde el usuario podrá agregar sus datos personales y registrar medidas de partes del cuerpo y sus prendas de vestir preferidas, además de publicar dicho perfil en la base de datos central del sistema, para que sus contactos, obtenidos de la agenda de contactos personal del dispositivo móvil, puedan consultarlo, siempre y cuando hayan aceptado la solicitud de amistad.

Se debe desarrollar un módulo de gestión de marcas, en el cual el aplicativo le dará posibilidades al usuario de definir cuáles son sus marcas de prendas de vestir favoritas, elegidas de una lista predefinida mostrada por el aplicativo.

Se debe desarrollar un módulo de carga, visualización de imágenes y tallas de prendas, en el cual se podrá realizar su búsqueda en **Amazon.com**, único proveedor deseado para el desarrollo del aplicativo móvil, obteniendo los registros que concuerden con los criterios deseados. Las tallas serán manejadas en pulgadas y centímetros, el aplicativo móvil realizará su conversión, permitiendo al usuario un registro de prendas y búsqueda versátil. También se manejará la opción de compra, en donde se establecerá una conexión directa con el producto de **Amazon.com** para que el usuario pueda realizar el pago en su plataforma.

Es importante para el desarrollo del aplicativo móvil tomar en cuenta que la interacción del usuario no se puede ver afectada por el sistema operativo que esté utilizando el dispositivo móvil, por ello el desarrollo de la interfaz gráfica de la solución no se verá influenciada por los estándares de desarrollo de interfaces graficas de *iOS* y *Android*, ya que se definirá una interfaz común para ambos sistemas operativos.

I.3 OBJETIVO GENERAL

Desarrollar un aplicativo móvil para consultar y compartir prendas de vestir mediante la utilización de estrategias de comparación por tallas.

I.4 OBJETIVOS ESPECÍFICOS

1. Diseñar e implementar un modelo de base de datos para que contemple la información requerida por la aplicación.
2. Desarrollar un módulo de Registro.
3. Desarrollar un módulo de Configuración inicial.
4. Desarrollar un módulo de Gestión de tallas y prendas.
5. Desarrollar un módulo de Gestión de marcas.
6. Desarrollar un módulo de Gestión de perfiles.
7. Desarrollar un módulo de Gestión de contactos.
8. Desarrollar un módulo de búsqueda, comparación y emparejamiento de prendas de vestir.
9. Desarrollar un módulo de Gestión de notificaciones.
10. Desarrollar un módulo de Configuraciones y ajustes.

I.4.1 Aporte Funcional

11. Diseñar los procesos involucrados en la búsqueda de prendas de vestir mediante comparaciones y emparejamientos con el uso de tallas como criterio a través del aplicativo móvil.

I.4.2 Aporte Tecnológico

12. Evaluar entre las distintas tecnologías para el desarrollo de plataformas móviles existentes, que brinden mayor escalabilidad y adaptabilidad de las distintas plataformas móviles.
13. Evaluar e implementar interoperabilidad entre las distintas tecnologías para la vinculación de aplicaciones móviles con *Cloud Storage*, notificaciones *push* y gestión de usuarios.
14. Implementar interoperabilidad con **Amazon.com**.
15. Implementar interoperabilidad con los servicios de **Facebook**.

I.5 ALCANCE

1- Diseñar e implementar un modelo de base de datos para que contemple la información requerida por la aplicación

El modelo de datos soportará la información requerida por la solución propuesta, permitiendo que los administradores de la base de datos puedan registrar los tipos de prendas de vestir y tallas a manejar en la aplicación. Además de contemplar toda la información referente a los usuarios, información de perfiles, publicaciones, notificaciones, resultados comparaciones y emparejamientos, contactos, prendas favoritas, medidas del cuerpo y tallas. Estos datos serán almacenados en una base de datos *NoSQL* alojada en la nube.

2- Desarrollar un módulo de Registro

Este módulo se encargará del registro de los usuarios en el aplicativo móvil, el cual se realizará de dos formas, la primera, mediante la solicitud de un código de verificación recibido a través de un mensaje de texto al número telefónico del usuario, y la segunda, mediante la autenticación con una cuenta en la red social **Facebook**.

3- Desarrollar un módulo de Configuración inicial

En éste módulo se manejará una inducción del usuario al aplicativo móvil, en la cual ingresará su género y la talla de una prenda en específico elegida de la lista de prendas de vestir del aplicativo móvil. El fin de este módulo es establecer la información base del usuario, para que pueda hacer uso de todo el funcionamiento del aplicativo móvil desde su primera interacción.

4- Desarrollar un módulo de Gestión de tallas y prendas

Este módulo se encargará de administrar el registro y modificación de las prendas preferidas del usuario. Además podrá crear y agregar nuevas tallas de prendas de vestir y medidas del cuerpo, donde las opciones que se listará en el aplicativo móvil serán: brazo, busto, pecho, dedos, pies, cabeza, cadera, pierna,

cuello, hombro, muslo y cintura. Al momento de agregar una medida de una parte del cuerpo, el aplicativo móvil habilitará una opción para ver un video en el cual se explicará la manera de medir la parte del cuerpo seleccionada.

5- Desarrollar un módulo de Gestión de marcas

Este módulo se encargará de listar marcas de prendas de vestir registradas en la aplicación y cada usuario tendrá la opción de seleccionar cuáles de ellas son sus preferidas, para luego publicar sus preferencias en sus perfiles.

6- Desarrollar un módulo de Gestión de perfiles

Este módulo se encargará del manejo de los perfiles de los usuarios. Cada usuario tendrá la opción de crear varios perfiles y publicarlos para distintos usuarios de la lista de contactos. En cada perfil se manejarán datos personales como nombre, fecha de nacimiento, genero, correo, medidas de partes del cuerpo y preferencias de distintos tipos de prendas de vestir.

7- Desarrollar un módulo de Gestión de contactos

Este módulo de encargará de gestionar las listas de contactos de los usuarios, además de aprobación, rechazo o envíos de solicitudes de amistad. El usuario podrá ver los perfiles de sus contactos, siempre y cuando estos estén publicados y puedan ser vistos por el usuario que los consulta.

8- Desarrollar un módulo de búsqueda, comparación y emparejamiento de prendas de vestir

Se propone diseñar e implementar este módulo con el fin de permitir al usuario, mediante el uso de la aplicación, obtener una lista de prendas de vestir que más se ajusten a sus especificaciones (precio y tallas).

Se desarrollará una aplicación encargada de realizar un *web scraping* en el sitio web de Amazon.com y de almacenar en una base de datos relacional el resultado del

recorrido del sitio web. Dicha aplicación será ejecutada en un servido local, donde al final de su ejecución enviará a persistencia las prendas de vestir encontradas.

Luego de que el usuario realice el proceso de registro de medidas del cuerpo, el aplicativo realizará un proceso de cálculo de tallas, en donde se hace una conversión a una talla específica en cada prenda, con base en las prendas de vestir preferidas y medidas del cuerpo registradas en el perfil del usuario que está realizando la búsqueda.

Luego de haber calculado las tallas relacionadas a las medidas del usuario, el aplicativo móvil, realizará una búsqueda de prendas en un servicio web que consultará la base de datos relacional previamente cargada con las prendas obtenidas en el proceso de web scraping, comparando con base en los datos mencionados anteriormente y rigiéndose por las configuraciones establecidas en el aplicativo, relacionadas con el nivel de exactitud de las comparaciones (rango de precios de las prendas y porcentaje de exactitud de comparación de las prendas en cuanto a tallas), teniendo como resultado un listado de las prendas consultadas que cumplieron con los criterios mencionados, mostrando imagen, marca y nombre de la prenda, precio y porcentaje de exactitud.

El usuario podrá realizar dos acciones con cada una de las prendas de vestir listadas como resultado de la ejecución de éste proceso, la primera es la opción de dar gustar (*like*) a la prenda, para poder registrarla y consultarla en un futuro, y la segunda es la opción de compra, donde luego de seleccionar ésta opción, el aplicativo móvil re-direccionará a **Amazon.com**, para realizar la compra desde su plataforma.

En caso de que el usuario quiera consultar en un futuro una prenda a la cual le dio gustar (*like*), y ésta fue eliminada del sitio web origen de la prenda, se mostrará una pantalla de alerta en la cual se le indicará al usuario que la prenda fue removida del sitio web.

Cada vez que el usuario actualice su perfil, agregando, eliminando o modificando medidas del cuerpo y prendas de vestir, el resultado del cálculo de tallas

será distinto, por lo tanto el resultado de la ejecución de proceso de búsqueda, comparación y emparejamiento lo será también.

9- Desarrollar un módulo de Gestión de notificaciones

Este módulo se comportará como un apartado de notificaciones, en el cual se listaran todas las solicitudes de amistad de otros usuarios, *likes* que reciba cada publicación realizada por el usuario poseedor de la cuenta. Se podrán recibir notificaciones como cumpleaños de un contacto, actualización de perfil de un contacto, una petición de tallas, aprobación una petición y si alguna prenda hace un match perfecto en una comparación.

10- Desarrollar un módulo de Configuraciones y ajustes

Este módulo se encargará de administrar las configuraciones del aplicativo móvil, con la posibilidad de gestionar las configuraciones referentes a los perfiles, notificaciones y unidad de medida por defecto (centímetros o pulgadas). El usuario tendrá la posibilidad de elegir si desea compartir o no su perfil. También se podrá configurar cuales tipos de notificaciones quiere recibir el usuario, donde tendrá las siguientes opciones: cuando algún contacto cumple años, actualiza su perfil, hace una petición de tallas, aprueba una petición y si alguna prenda hace un match perfecto en una comparación. Además un usuario podrá limitar el nivel de exactitud de las comparaciones en las búsquedas de prendas de vestir, en cuanto a rango de precios y tallas.

Aporte Funcional

11- Diseñar los procesos involucrados en la búsqueda de prendas de vestir mediante comparaciones y emparejamientos con el uso de tallas como criterio a través del aplicativo móvil

La consulta de las prendas de vestir en **Amazon.com** actualmente es muy abierta, es decir, existen numerosos filtros de búsquedas que pueden ser utilizados por las personas, haciendo engorrosa la ubicación de sus preferencias. Estos

resultados pueden arrojar un gran número de prendas que no forman parte de las especificaciones iniciales indicadas.

Se deberá diseñar un método que permita que las consultas de las prendas de vestir se ajusten a las especificaciones de los usuarios (precios, tallas y tipos de prendas de vestir preferidas), aplicando comparaciones con estos parámetros. De este modo el usuario podrá tener una búsqueda más acertada de las prendas que más se ajustan a sus especificaciones, obteniendo las mejores opciones encontradas en los sitios webs proveedores.

Aporte Tecnológico

12- Evaluar entre las distintas tecnologías para el desarrollo de plataformas móviles existentes, que brinden mayor escalabilidad y adaptabilidad de las distintas plataformas móviles

Se deberán evaluar las distintas alternativas de *frameworks* de desarrollo de aplicaciones multiplataforma existentes, que permitan desarrollar aplicaciones con la mayor cantidad de herramientas genéricas, para así cumplir con todos los requerimientos planteados por *Trascend Software Evolution.* Se deberán evaluar las facilidades ofrecidas por cada una de las alternativas en torno a la interacción con los componentes de los dispositivos móviles.

El uso de este tipo de *framework* permite que el aplicativo a desarrollar se pueda utilizar en la gran mayoría de los dispositivos móviles, ya que este será embebido en el navegador de aplicaciones nativas de cada dispositivo.

13- Evaluar e implementar interoperabilidad entre las distintas tecnologías para la vinculación de aplicaciones móviles con *Cloud Storage*, notificaciones *push* y gestión de usuarios

Se deberán evaluar las distintas alternativas de tecnologías que permitan las implementación de un modelo de distribución de *Software* llamado "*Software as a Service*", donde el soporte lógico y los datos está alojados en un servidor externo perteneciente al proveedor de este tipo de servicios.

El uso de este tipo de tecnologías elimina la necesidad de instalar y ejecutar aplicaciones en computadores, servidores y centros de datos propios, además de eliminar el costo de mantenimiento.

14- Implementar interoperabilidad con Amazon.com

Se implementará la interoperabilidad con Amazon.com para la obtención de los datos de los productos consultados (precio, talla, imagen, marca, nombre, tipo de prenda) a través del proceso de consulta de prendas de vestir del aplicativo móvil empleando la técnica de *web scraping* a través de una aplicación.

15- Implementar interoperabilidad con los servicios de Facebook

Se implementará la interoperabilidad con los servicios de autenticación y obtención de datos de usuarios (nombre y fecha de nacimiento), ofrecidos por **Facebook**, para ser utilizados en el proceso de registro de usuarios en el aplicativo móvil.

I.6 LIMITACIONES

- El aplicativo móvil será desarrollado mediante un *framework* de desarrollo de aplicaciones móviles híbridas, por lo tanto el mismo va dirigido a usuarios de dispositivos móviles que posean sistema operativo *iOS* y *Android*.
- El uso de la aplicación móvil será permitido solo si el usuario común fue previamente registrado en el sistema, ya que la aplicación móvil requiere de esta información para ciertos módulos.
- El aplicativo móvil será soportado por *iOS* desde su versión 6+ y por *Android* desde su versión 4.0+.
- La aplicación de *web scraping* será ejecutada en un servidor local.
- El idioma por defecto del aplicativo móvil será el inglés.

I.7 JUSTIFICACIÓN

En los últimos años se ha ido incrementando la dependencia de la sociedad respecto a los *smartphones* y, con esto, a los aplicativos móviles, llegado casi al nivel de indispensabilidad para el desarrollo de la vida normal de una persona.

Actualmente la búsqueda de prendas de vestir en los sitios web proveedores de este tipo de productos se ha vuelto muy engorrosa y en ocasiones se obtienen resultados que no concuerdan o que no poseen ningún tipo de relación con las especificaciones definidas previo a la búsqueda.

El presente Trabajo Instrumental de Grado, tiene su justificación en la necesidad que actualmente posee **Trascend Software Evolution** de desarrollar un aplicativo móvil que permita a sus usuarios consultar prendas de vestir en Amazon.com, obteniendo como resultado las prendas alojadas en éste sitio web que más se adapten a sus especificaciones (tallas y rango de precios). Este aplicativo permitirá a sus usuarios ubicar con más facilidad sus prendas de vestir deseadas y ajustadas a las medidas de sus cuerpos.

CAPÍTULO II – MARCO TEÓRICO

II.1 SISTEMAS OPERATIVOS

Un sistema operativo es el software principal que sirve de intermediario entre un dispositivo y el usuario, y son los encargados de administrar los recursos, coordinar el hardware y organizar archivos y directorios en dispositivos de almacenamiento. [1]

II.1.1 Sistemas Operativos Móviles

Los sistemas operativos móviles son un tipo de sistema operativo, pero más simples, orientados a la conectividad inalámbrica, los formatos multimedia para móviles y las diferentes maneras de introducir información entre ellos. [2]

A continuación se listan los sistemas operativos móviles más populares de los últimos años, según IDC: [3]

II.1.1.1 Sistema Operativo Android

Es un sistema operativo móvil basado en Linux, que junto con aplicaciones de middleware está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes (*smartphones*), tabletas (*tablets*), entre otros dispositivos. [2]

Fue desarrollado inicialmente por **Android Inc.**, una firma comprada por **Google** en 2005. Es el principal producto de la **Open Handset Alliance**, un conglomerado de fabricantes y desarrolladores de hardware, software y operadores de servicio. [2]

Android se encuentra actualmente en su versión 7.0, denominada Nougat. [3]

Andriod permite a los fabricantes de los dispositivos la personalización de la interfaz de usuario, con la finalidad de que estos, bajo su propio punto de vista, provean un manejo más simple del sistema. Por ello en el mercado existen dos tipos de dispositivos Android, los puros (no personalizados) y los personalizados. [4]

Según estudios realizados luego de la culminación del tercer trimestre del año 2016, Android se ubica en el primer puesto del mercado de *smartphones* a nivel mundial, obteniendo un 86,8%. [3]

II.1.1.2 Sistema Operativo iOS

Es un sistema operativo móvil de **Apple, Inc**, originalmente desarrollado para el *iPhone*, siendo después usado en dispositivos como el *iPod Touch*, *iPad* y el *Apple TV*. iOS se deriva de Mac OS X, que a su vez está basado en Darwin BSD, y por lo tanto es un sistema operativo Unix. Se encuentra actualmente en su versión 10. [5]

La interfaz de usuario de iOS está basada en el concepto de manipulación directa, usando gestos multitáctiles. Los elementos de control consisten de deslizadores, interruptores y botones. La respuesta a las órdenes del usuario es inmediata y provee una navegación fluida. La interacción con el sistema operativo incluye gestos como deslices, toques, pellizcos, los cuales tienen definiciones diferentes dependiendo del contexto de la interfaz. [5]

Según estudios realizados luego de la culminación del tercer trimestre del año 2016, iOS se ubica en el segundo puesto del mercado de *smartphones* a nivel mundial, obteniendo un 12,7%. [3]

A continuación, en la Figura 1. – Mercado mundial de SO de *smartphones*, se muestran los porcentajes de ventas desde el tercer trimestre de 2013 hasta el tercer trimestre de 2016.

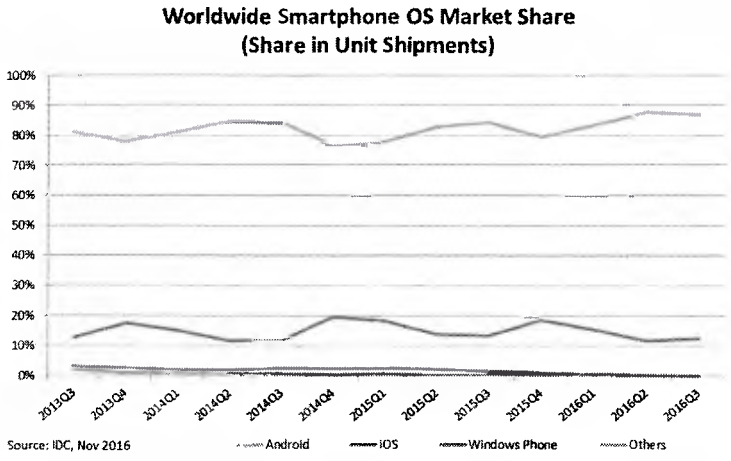

Figura 1. – Mercado mundial de SO de *smartphones.* Fuente: Smartphone OS Market Share, 2016 Q3 (2016).

II.2 APLICACIONES MÓVILES

Una aplicación móvil es un *software* desarrollado para ser ejecutado a través de un dispositivo móvil inteligente, tableta (*tablet)* u otro. Su distribución se basa en tiendas virtuales dedicadas al sistema operativo que del dispositivo. [6]

II.2.1 Aplicaciones Nativas

Las aplicaciones nativas son aquellas desarrolladas para un sistema operativo en específico, empleando las plataformas y lenguajes destinados para éste. El desarrollo nativo permite utilizar al máximo los recursos del dispositivo (tanto del sistema operativo como del *hardware*). Una desventaja de este tipo de aplicaciones móviles es precisamente que están destinadas a un sistema operativo, esto implica que el desarrollo para distintas plataformas requiere de un costo elevado en cuanto al tiempo de desarrollo. [6]

II.2.2 Aplicaciones *Web*

Las aplicaciones web son aquellas desarrolladas bajo lenguajes de programación y *frameworks* de desarrollo web. Este tipo de aplicaciones pueden ser distribuidas en distintos sistemas operativos ya que sólo requieren de un navegador

web para ser ejecutadas. Las aplicaciones web no son publicadas en las plataformas de distribución de software móvil. [6]

II.2.3 Aplicaciones Híbridas

El enfoque híbrido combina el desarrollo nativo con tecnología web. Los desarrolladores escriben gran parte del código de su aplicación en tecnologías web para múltiples plataformas, y mantienen el acceso directo a API (Interfaz de Programación de Aplicaciones) nativas cuando lo necesitan. [7]

La porción nativa de la aplicación emplea APIs de sistemas operativos para crear un motor de búsqueda HTML incorporado que funcione como un puente entre el navegador y las APIs del dispositivo, permitiendo que la aplicación híbrida aproveche todas las características que ofrecen los dispositivos modernos. [7]

La porción web de la aplicación puede ser una página web que resida en un servidor o bien un conjunto de archivos HTML, JavaScript, CSS y medios, incorporados en el código de la aplicación, almacenados localmente en el dispositivo, siendo la segunda opción la más utilizada, ya que mejora el desempeño y permite la disponibilidad *offline*. [7]

La utilización de este enfoque para el desarrollo de aplicativos móviles permite reducir el tiempo de codificación y aumentar la disponibilidad, ya que el desarrollo va dirigido a varios sistemas operativos móviles. Esto se logra creando un núcleo de código común utilizando tecnologías web y encapsulándolo en un contenedor específico para cada plataforma, de manera que la aplicación interactúe con el sistema operativo adaptándose a sus características y permitiendo acceder al hardware del dispositivo. Además, permite la distribución en las tiendas oficiales de cada plataforma. [7]

II.3 LENGUAJES DE PROGRAMACIÓN

Un lenguaje de programación es un idioma artificial diseñado para expresar computaciones o instrucciones que pueden ser llevadas a cabo por dispositivos como

computadoras, móviles, entre otros. Permiten llevar a cabo la interacción humano computador, transformando sentencias de un lenguaje comprendido por los humanos a uno comprendido por el computador (lenguaje máquina). Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. [8]

II.3.1 JavaScript

Es un lenguaje ligero e interpretado, orientado a objetos, conocido como el lenguaje de script para páginas web. JavaScript se basa en los estándares del lenguaje ECMAScript, manejado por la *European Computer Manufacturers Association* (ECMA). [11]

II.3.2 Java

Es un lenguaje de programación multiplataforma orientado a objetos desarrollado por James Gosling y publicado por primera vez por Sun Microsystems, compañía adquirida tiempo después, específicamente en el año 2010, por Oracle. [12]

II.4 FRAMEWORKS DE DESARROLLO MÓVIL

II.4.1 Apache Cordova

Framework de desarrollo móvil de código abierto. Permite utilizar tecnologías *web* estándar HTML5[5], CSS3[6] y JavaScript para el desarrollo multiplataforma. Las aplicaciones se ejecutan dentro de paquetes dirigidos a cada plataforma y utilizan *plugins* para acceder a las capacidades de cada dispositivo, como sensores, datos, estado de la red, entre otros. [23]

---

[5] **HTML**: Es el lenguaje base que se emplea para el desarrollo de páginas web. Se usa para crear y representar visualmente una página web, estableciendo el contenido pero no su funcionalidad. [9]

[6] **CSS**: Es uno de los lenguajes base de la Open Web y posee una especificación estandarizada por parte del W3C. Es el lenguaje utilizado para describir la presentación de documentos HTML. Describe como debe ser renderizado el elemento estructurado en pantalla. [10]

II.4.2 jQuery Mobile

Framework JavaScript para el desarrollo rápido y fácil de sitios webs optimizados para teléfonos móviles. Con JQuery Mobile se evita conocer la lógica específica de cada dispositivo y se centra en la programación para un solo fin, el navegador de un teléfono móvil. [24]

II.4.3 Xamarin

Framework de desarrollo de aplicaciones móviles que permite escribir el código de las aplicaciones en el lenguaje C# para traducirlo con la finalidad de ejecutarse en iOS, Android y Windows Phone. [25]

Ésta alternativa de desarrollo móvil permite reutilizar el código de toda la lógica de la aplicación, sin embargo esto no sucede con la interfaz gráfica, dado que está atada a los estándares nativos de cada plataforma. [26]

II.5 BASES DE DATOS

Una base de datos es un centro de almacenamiento de información que permite mantener los datos organizados y manipularlos (agregar, eliminar y actualizar) con la finalidad de obtener información estructurada en base a ellos (consultas). [13]

II.5.1 Bases de datos relacionales

Una base de datos relacional es una base de datos tratada a través de un conjunto de tablas y manipulada de acuerdo a un modelo de datos relacional. Existen distintos tipos de objetos manejados en este tipo de base de datos, tales como tablas, índices, funciones, vistas, entre otros. El lenguaje base para la manipulación de los datos y objetos es SQL[7]. [14]

[7] **SQL o *Structured Query Language***: es un lenguaje que da acceso a un sistema de gestión de bases de datos relacionales que permite especificar diversos tipos de operaciones en ellos.

II.5.1.1 Gestores de bases de datos relacionales más usados

- MySQL: Es un manejador de base de datos relacional, multihilo y multiusuario que ofrece distintas ventajas, como por ejemplo, velocidad al realizar las operaciones, bajo costo en requerimientos para la elaboración de bases de datos, facilidad de configuración e instalación. Actualmente lidera el mercado de bases de datos relacionales de código abierto. [15]
- Oracle: Es uno de los sistemas de gestión de base de datos relacional más fiable y usado. Creado por Oracle Corporation, una de las compañías más grandes en el ámbito tecnológico. Generalmente es empleada para grandes proyectos y empresas debido a su gran potencia y alto costo. [16]
- SQL Server: Es un sistema de gestión de bases de datos relacionales creado por Microsoft, diseñado para el entorno empresarial. Se ejecuta en T-SQL (Transact – SQL), un conjunto de extensiones de programación que añaden varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas. [17]

II.5.2 Bases de datos NoSQL

Son sistemas de almacenamiento de información que no cumplen con el esquema relacional. Tampoco utilizan una estructura de datos en forma de tabla para el almacenamiento sino que hacen uso de otros formatos como clave–valor, mapeo de columnas o grafos para ello. [18]

A continuación se describen los tipos de bases de datos NoSQL: [18]

II.5.2.1 Clave – Valor

Cada elemento está identificado por una llave única, lo que permite la recuperación de la información de forma muy rápida, información que habitualmente está almacenada como un objeto binario (BLOB).

II.5.2.2 Orientadas a documentos

Este tipo de base de datos almacena la información como un documento, generalmente utilizando una estructura JSON o XML. Este tipo de implementación permite, además de realizar búsquedas por clave–valor, realizar consultas más avanzadas sobre el contenido del documento.

II.5.2.3 Orientadas a grafos

La información se representa como nodos de un grafo y sus relaciones con las aristas del mismo, de manera que para recorrerlas se hace uso de la teoría de grafos. Para sacar el máximo rendimiento a este tipo de bases de datos, su estructura debe estar totalmente normalizada.

II.6 SERVICIOS WEB

Son un conjunto de aplicaciones o de tecnologías con capacidad para inter-operar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la *Web*. [19]

II.6.1 Servicios web RESTful

Son servicios web que implementan la arquitectura Transferencia de Estado Representacional (REST) en conjunto con el protocolo HTTP. Esta arquitectura define una serie de reglas que toda aplicación REST debe cumplir: [20]

- Arquitectura cliente-servidor: consiste en una separación clara y concisa entre los dos agentes básicos en un intercambio de información: el cliente y el servidor. Agentes que deben ser independientes entre sí. [20]
- Sin estado (*Stateless*): El servidor no puede identificar si una petición es de un mismo cliente o de uno diferente, por tanto el servidor trata cada petición como una nueva. [20]

- Cacheable: esta norma implica que el servidor que sirve las peticiones del cliente debe definir algún modo de capturar dichas peticiones, para aumentar el rendimiento, escalabilidad, entre otros. [20]
- Sistema por capas: un sistema no debe forzar al cliente a saber por cuál(es) capas se tramita la información, lo que permite que el cliente conserve su independencia con respecto a dichas capas. [20]
- Interfaz uniforme: no importa quién haga las peticiones ni quien las reciba, siempre y cuando ambos cumplan una interfaz definida de antemano. [20]

Un servicio web RESTful debe estar conformado por:

- Identificador Uniforme de Recursos (URI): dirección de acceso al recurso del servicio que se desee hacer la petición. [20]
- El tipo de representación de dicho recurso: Características del recurso que pueden ser definidas en la cabecera de la petición, por ejemplo: "*Content-type: application/json*", establece que la respuesta del servidor será en formato JSON. El tipo es arbitrario, siendo los más comunes XML[8], JSON [9]y TXT[10]. [20]
- Operaciones soportadas: HTTP define varios tipos de operaciones, donde los más usado son *GET* (obtener), *POST* (enviar), *PUT* (modificar) y *DELETE* (eliminar). [20]
- Hipervínculos: la respuesta puede incluir hipervínculos hacia otras acciones que podamos realizar sobre los recursos. [20]

[8] **XML o *eXtensible Markup Language***: permite definir lenguajes de marcas desarrollado por el W3C, utilizado para almacenar datos en forma legible.
[9] **JSON o *JavaScript Object Notation***: formato de texto ligero para el intercambio de datos.
[10] **TXT o archivo de texto**: formato que estructura series de líneas de texto.

II.7 BACKEND AS A SERVICE (BAAS)

BaaS es un enfoque para proporcionar a los desarrolladores de aplicaciones web y móviles una forma de conectar sus aplicaciones con un *backend*[11] para el almacenamiento y procesamiento en la nube, además de proporcionar características comunes como administración de usuarios, notificaciones *push*[12], integración de redes sociales y otras características que los usuarios móviles demandan de sus aplicaciones. Estas características son proporcionadas a través de herramientas (kits) de desarrollo de software (SDKs) personalizados y APIs. [21]

Este enfoque brinda las siguientes ventajas [21]:

- Aumento de la eficiencia: Reduciendo la sobrecarga en todos los aspectos del desarrollo de aplicaciones móviles, aumentando la eficiencia en todas las etapas del desarrollo.
- Rápidos tiempos de mercado: Reduciendo los obstáculos para llevar una aplicación móvil desde la idea hasta la producción y sobrecarga con operaciones una vez en producción.
- Entrega de aplicaciones con menos recursos: BaaS apoya el desarrollo con menos desarrolladores y soporte de datos y recursos de Tecnologías de Información (TI).
- Optimizar para móviles y *tablets*: los proveedores de BaaS han dedicado mucho tiempo y recursos en optimizar los datos y la red para aplicaciones móviles, con la finalidad de reducir los problemas de fragmentación en múltiples plataformas y dispositivos.

---

[11] ***Backend***: es la sección de sistema que se encuentra del lado del servidor y se encarga de interactuar con bases de datos, verificar manejos de sesiones de usuarios, entre otras acciones solicitadas por los usuarios del sistema.

[12] **Notificaciones *push***: mensajes que se envían de forma directa a los dispositivos móviles. Su uso va de la mano con noticias ligadas a las aplicaciones instaladas en los dispositivos.

- Infraestructura segura y escalable: BaaS proporciona una infraestructura integrada que se ocupa de escalabilidad, seguridad, rendimiento y otras funcionalidades operacionales, facilitando el desarrollo de *software*.

II.8 PLATAFORMAS DE *CLOUD COMPUTING*

II.8.1 Google Firebase

Es una plataforma de desarrollo móvil en la nube ofrecida por Google y disponible para diferentes plataformas (Android, iOS y Web). Firebase ofrece variadas funcionalidades para facilitar el desarrollo de aplicaciones. Ofrece bases de datos en tiempo real, autenticación de usuarios, mensajería en la nube, análisis de datos (*Analytics*), notificaciones *push*, entre otras funcionalidades. [22]

Firebase Realtime Database es una base de datos NoSQL alojada en la nube orientada a documentos, donde los datos se almacenan en formato JavaScript Object Notation (JSON) y se sincronizan en tiempo real con cada usuario conectado. Todos los usuarios comparten una instancia de *Realtime Database* y reciben actualizaciones de forma automática con los datos más nuevos. Esta base de datos puede integrarse con los lenguajes de programación Java, Objetive-C, JavaScript y Swift. [23]

Actualmente, ofrece una herramienta llamada AngularFire, que permite su manipulación a través del *framework* de desarrollo de aplicaciones web AngularJS[13], facilitando la interacción de aplicaciones web con ésta plataforma.

II.8.2 Amazon Web Services (AWS)

Es una plataforma de servicios en la nube que ofrece potencia de cómputo, almacenamiento de datos, entrega de contenido, entre otras funcionalidades importantes para el buen desempeño de las aplicaciones. [24]

Ofrece una serie de *kits* de desarrollo (AWS SDK), disponibles para numerosos lenguajes de programación, entre ellos, Java, .Net, PHP, Python y Javascript. [25]

---

[13] **AngularJS**: es un *framework* de JavaScript de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página.

Posee una gran cantidad de componentes ofrecidos a los clientes, de los cuales destacan Amazon DynamoDB y Amazon SNS.

II.8.2.1 Amazon DynamoDB

Servicio de base de datos NoSQL ofrecido por *Amazon Web Services* (AWS). Rápido y flexible para todas las aplicaciones que requieren latencias constantes y de meros milisegundos a cualquier escala. Es una base de datos totalmente administrada en la nube, compatible con modelos de almacenamiento de datos de valor/clave y de documentos. Posee un flexible modelo de datos y un buen desempeño, lo que la convierte en una herramienta ideal para aplicaciones móviles, web, juegos, entre otros. [26]

Una de las ventajas competitivas que posee esta herramienta es la despreocupación por el mantenimiento y la administración de la base de datos, ya sea componentes físicos (*hardware*) o intangibles (*software*). [27]

Posee la opción de sincronizarse con el servicio de respuesta impulsada por eventos, proporcionada por Amazon como parte de los Servicios Web de Amazon (AWS Lambda), lo que permite ejecutar un segmento de código luego de realizar alguna modificación en una de las tablas creadas alertado por disparadores (*triggers*). [28]

DynamoDB está integrada con un manejador que gestiona la creación de varios usuarios, AWS *Identity and Access Management* (IAM), permitiendo manejar el control de acceso de usuarios, asignando credenciales de acceso al servicio y recursos a través de una cuenta de Amazon Web Services. [27]

II.8.2.2 Amazon SNS

Es un servicio de notificaciones móviles y mensajes de publicación/suscripción completamente administrado para coordinar la entrega de mensajes a clientes y puntos de conexión suscritos. Sus servicios son ofrecidos a dispositivos basados en iOS, Android, Fire OS, Windows, entre otros. [30]

II.9 WEB CRAWLER Y WEB SCRAPING

Son aplicaciones encargadas de recorrer sitios *web*, obteniendo información y contenidos publicados en ellos. Comienzan sus procesos ingresando en páginas webs predeterminadas y a partir de esto analizan los enlaces que éstas poseen, para luego ingresar en los mismos. Implementan una técnica llamada web *scraping*, empleada para extraer información de las páginas *web*, analizando cada una de las etiquetas que éstas contienen. [31]

CAPÍTULO III – MARCO METODOLÓGICO

En el presente Trabajo Instrumental de Grado se aplicó el marco de trabajo Scrum, el cual permite tener una comunicación constante con el cliente y además está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto; los requisitos son cambiantes o poco definidos y donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales. [32]

III.1 FASES

Existen cuatro fases bien definidas que integran el ciclo de vida del producto bajo el marco de trabajo Scrum: [33]

- *Continuous Planning and Backlog Grooming* (Planificación Continua y Refinamiento del *Backlog*): en esta fase se hace el refinamiento del *product backlog* y del *sprint backlog*, normalmente se hace luego de la finalización de cada iteración (*sprint*) y una vez al inicio del ciclo de vida del producto. Se organizan y priorizan los requerimientos con el fin de establecer un orden de realización. El principal entregable de esta fase es el *product backlog* y el *sprint backlog*.
- *Sprint*: en esta fase se procede a desarrollar los requerimientos listados en el *sprint backlog* correspondiente. La duración de un *sprint* es de 1 a 4 semanas, exceptuando circunstancias especiales. Además diariamente, durante el *sprint*, se realiza una reunión con una duración de 15 minutos, para analizar el trabajo y las dificultades que se pudieron haber tenido en el día anterior. También el *project manager* organiza reuniones semanales, no para evaluar el estatus del proyecto sino para evaluar el progreso del equipo hacia el objetivo del *sprint*. El principal entregable de esta fase son los requerimientos desarrollados.
- Inspección y Adaptación: esta fase consta de una reunión con el cliente, en la cual se revisa el *product increment*, se determinan nuevos requerimientos (en

caso de ser necesario) y se actualizan el *product backlog* y el *sprint backlog*. El principal entregable de esta fase es el *product backlog* y *sprint backlog* actualizados.

- Cierre: en esta fase se realiza la reunión final con el cliente. El principal entregable de esta fase es el producto final.

En la Figura 2. Fases Scrum, se muestran gráficamente las fases explicadas anteriormente.

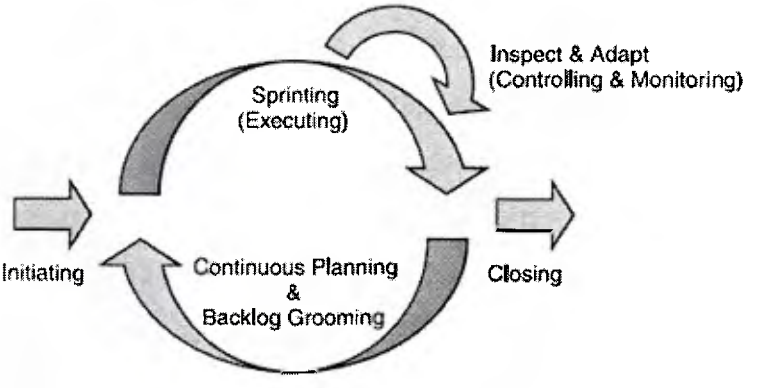

Figura 2. – Fases Scrum. Fuente: *Scrum in Action - Cengage Learning* (2012).

III.2 ARTEFACTOS

- *Product Backlog*: Es una lista ordenada que inicialmente contiene los requerimientos de lo que se conoce y entiende al momento de iniciar el desarrollo. Luego, el *product backlog* incluye todas las características, funciones, requerimientos, mejoras y correcciones que constituyen los cambios que deben introducirse en el producto en futuras versiones.
- *Sprint Backlog*: Es el conjunto de elementos seleccionados del *product backlog* para el *sprint*, además de un plan para la entrega del Incremento y el cumplimiento del objetivo del *sprint.*
- *Product Increment*: Es la suma de todos los elementos completados durante el desarrollo de los *sprints.* [34]

III.3 ROLES

Los roles involucrados en el desarrollo del aplicativo móvil son: [33]

- *Product Owner*: Representante de los accionistas y clientes que usan el *software*. Se focaliza en la parte de negocio y es el responsable del ROI del proyecto (entregar un valor superior al dinero invertido).
- *Scrum Master*: Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología.
- *Developer*: Encargado de desarrollar el aplicativo móvil.

El proyecto "Aplicativo móvil para consultar y compartir prendas de vestir mediante la utilización de estrategias de comparación por tallas", está integrado por un equipo de trabajo perteneciente a la empresa **Trascend Software Evolution**, conformado por un gerente de proyecto, un líder técnico y un desarrollador. Todo esto con el propósito de cumplir todos los requerimientos y desarrollar cada uno de los módulos que abarca la aplicación móvil.

III.4 OBJETIVOS Y SPRINTS

En la Figura 3. – Mapa de objetivos y *sprints*, se muestra gráficamente la ejecución del marco de trabajo seleccionado para el desarrollo del sistema final, siguiendo sus lineamientos y fases. Se muestran los módulos desarrollados en los *sprints* y sus relaciones con los objetivos y aportes establecidos para el cumplimiento de los requerimientos.

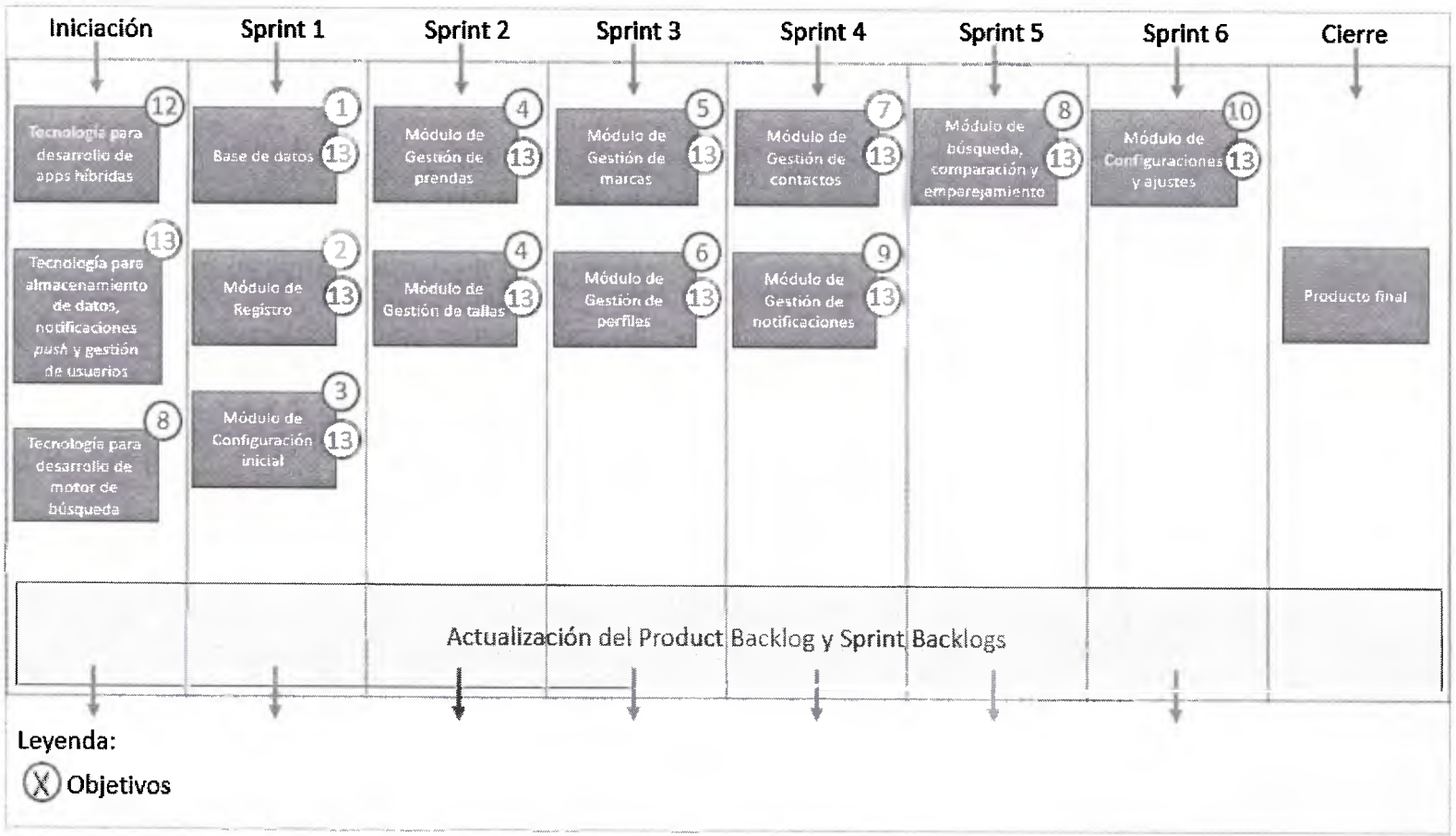

Figura 3. – Mapa de objetivos y sprints [Elaboración propia]

CAPÍTULO IV - DESARROLLO

IV.1 INTRODUCCIÓN

El desarrollo del aplicativo móvil se realizó dividiendo el trabajo en seis iteraciones (*sprints*), cumpliendo y acatando las normas definidas en el marco de trabajo seleccionado (**Scrum**).

En la Figura 4. – Flujo del sistema final, se representa el comportamiento esperado del sistema final desarrollado, indicando los componentes que posee, actores y los tipos de plataforma que emplea.

El administrador es el encargado de ejecutar el motor de búsqueda (*Web crawler*), donde cada prenda de vestir conseguida es almacenada en una base de datos relacional. Por su parte, el usuario del aplicativo móvil, consulta las prendas de vestir previamente cargadas por el motor de búsqueda. Los datos relacionados estrictamente con el aplicativo móvil, como registros de usuarios, medidas de partes del cuerpo, tipos de prendas de vestir favoritas, marcas favoritas y configuraciones, son almacenadas en una base de datos NoSQL proporcionada por una plataforma BaaS, permitiendo realizar envío y recepción de data en tiempo real. Las distintas notificaciones recibidas en el aplicativo móvil son enviadas a través de una sección habilitada por la plataforma BaaS para la gestión de notificaciones *push*. Para el registro de usuarios, se hace uso de los servicios ofrecidos por una plataforma que permita la utilización de mensajería en la nube, para la recepción de mensajes de texto con un código de verificación, con el fin de realizar la validación del *smartphone* en el que se ejecuta el aplicativo móvil.

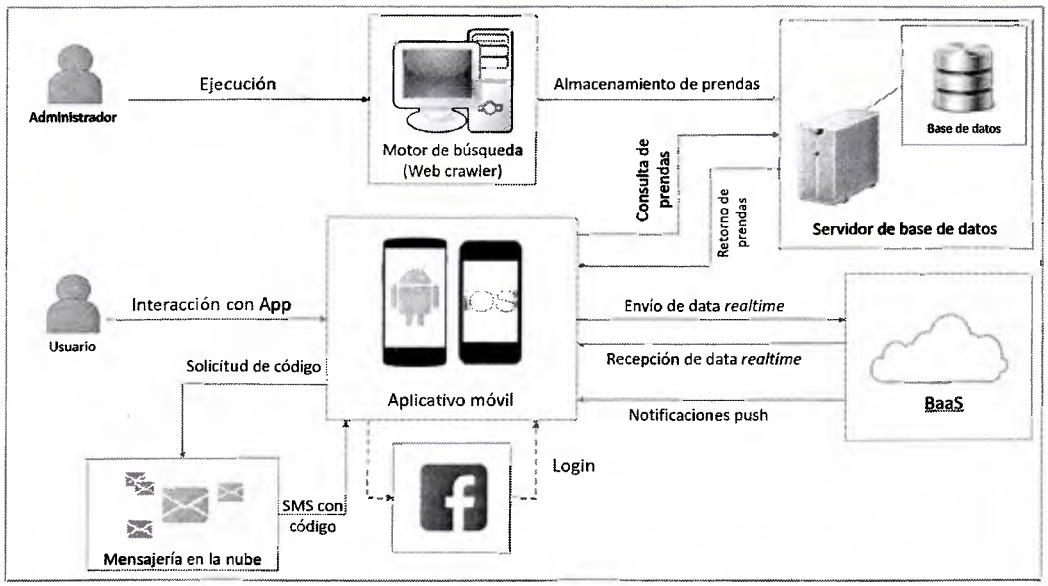

Figura 4. – Flujo final del sistema [Elaboración propia]

En la Tabla 1. – Objetivos desarrollados por iteración, se detallan los resultados de la ejecución de cada *sprint* ejecutado en cuanto al cumplimiento de los objetivos establecidos al principio de desarrollo de sistema.

Tabla 1. – Objetivos desarrollados por iteración [Elaboración propia]

Sprint	Objetivos Cumplidos	Entregables
Previo	8, 12, 13	Product Backlog Sprint Backlog
1	1, 2, 3, 13, 15	Incremento
2	4, 13	Incremento
3	5, 6, 13	Incremento
4	7, 9, 13	Incremento
5	8, 11, 13, 14	Incremento
6	10, 13	Producto Final

IV.2 ARQUITECTURA DEL SISTEMA

Para el desarrollo del sistema se optó por el uso de una arquitectura basada en capas, permitiendo una distribución de roles y responsabilidades de forma jerárquica bien definida. La arquitectura del sistema está conformada por las siguientes capas:

- **Capa de datos:** constituida por la base de datos relacional utilizada y la lógica que permite realizar las distintas operaciones necesarias sobre los datos almacenados en ella (consultas, inserción, actualización y eliminación).
- **Capa de servicios web:** conformada por los servicios web desarrollados para el acceso a los datos manejados por la capa de datos.
- **Servicios en la nube:** conformada por la plataforma de *cloud computing*, utilizada para el almacenamiento en la nube y notificaciones *push*.
- ***Web crawler*:** motor de búsqueda de prendas de vestir en **Amazon.com**. Utiliza la capa de servicios web para el almacenamiento de las prendas de vestir conseguidas.
- **Aplicativo móvil:** producto dirigido al usuario final que contempla todas las funcionalidades requeridas. Hace uso de la capa de servicios web para consultar las prendas de vestir registradas y de los servicios en la nube para el resto de las funcionalidades implementadas.

En la Figura 5. – Arquitectura del sistema, se muestra gráficamente como está constituido el sistema y la conexión entre las capas descritas anteriormente.

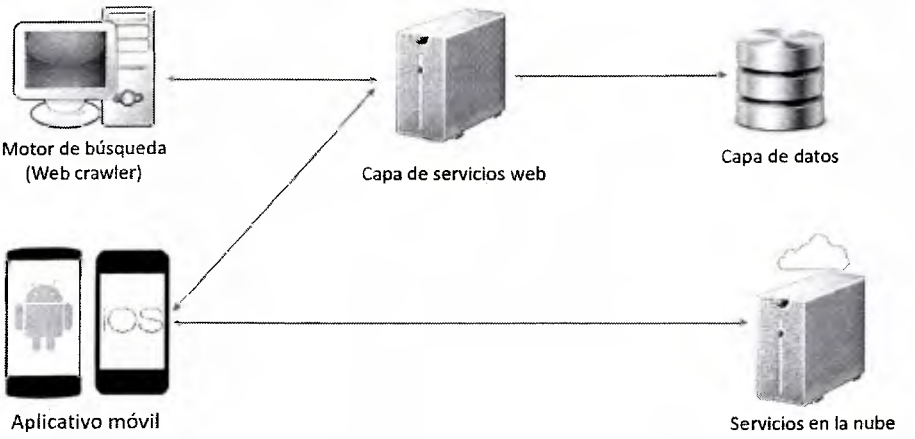

Figura 5. – Arquitectura del sistema [Elaboración propia]

IV.3 SELECCIÓN DE TECNOLOGÍAS PARA EL DESARROLLO DEL SISTEMA

IV.3.1 Selección de tecnología para el desarrollo del aplicativo móvil

Dada la necesidad de una aplicación apta para varias plataformas móviles (Android y iOS), se opta por tecnologías de desarrollo para aplicaciones híbridas. Las herramientas de desarrollo híbrido permiten construir sistemas bajo un mismo lenguaje, facilitando un motor de compilación integrado para cada sistema operativo móvil al que va dirigido (generalmente Android, iOS y Blackberry OS); gracias a éste servicio, se reducen los tiempos de trabajo, abarcando mayor campo y escalabilidad para un sistema.

Se requiere de un aplicativo móvil funcional para las plataformas Android y iOS. Los usuarios finales ejecutarán el aplicativo en *smartphones*, debido a que existe la necesidad de acceder a características que estos dispositivos poseen, tales como galerías de imágenes, registros de mensajes de texto (SMS) y agendas telefónicas.

Dada la necesidad de crear un aplicativo móvil funcional en varias plataformas, se considera la búsqueda de una herramienta que permita el desarrollo híbrido funcional multiplataforma; por ello, se toman en cuenta las herramientas (*frameworks*) de desarrollo que lideran el mercado en el campo tecnológico híbrido: Apache Cordova, JQuery Mobile y Xamarin.

Para la selección del *framework* a utilizar para el desarrollo del aplicativo móvil, se analizaron criterios en base a características indispensables con las que debe cumplir, asignando un punto al *framework* que cumpla con la característica evaluada y cero en caso contrario.

Dada la cantidad de herramientas que pueden satisfacer las necesidades del sistema, es de importancia seleccionar las adecuadas para acercarse al resultado final basado en las siguientes características:

- Costo: Valor monetario que supone el desarrollo del aplicativo móvil con el uso del *framework* evaluado. Se requiere que el *framework* seleccionado no implique costo en el desarrollo.
- Escalabilidad: Propiedad de un sistema que indica su habilidad para adaptarse a cualquier cambio, sin perder su calidad.
- Mantenimiento: Característica que posee un sistema para permitir a administradores realizar tareas para mantener su funcionamiento eficaz.
- Acceso a datos y características de dispositivos: Capacidad de un sistema de manipular los datos y *hardware* de un dispositivo, como cámara, almacenamiento, agenda telefónica, entre otros.

En la Tabla 2. – Comparación entre Frameworks de desarrollo híbrido, muestra el resultado obtenido de la evaluación de los *frameworks* para el desarrollo del aplicativo móvil según las características modelos a seguir.

Tabla 2. – Comparación entre Frameworks de desarrollo híbrido [Elaboración propia]

	Apache Cordova	Xamarin	jQuery Mobile
Costos	1	0	1
Escalabilidad	1	1	1
Mantenimiento	1	1	1
Acceso al hardware de dispositivos	1	1	0
Total	4	3	3

Xamarin presenta una desventaja para el desarrollo de este proyecto y es que está atada al pago de una licencia, lo que aumenta el costo del desarrollo del aplicativo móvil. Por otra parte, este proyecto presenta la necesidad de acceder a datos y características del dispositivo desde el aplicativo móvil y para ello se requieren de APIs o *plugins* que jQuery Mobile no provee, por lo tanto, éste *framework* no califica como elegible.

Por lo descrito, el *framework* de desarrollo móvil híbrido seleccionado fue Apache Cordova, cumpliendo las características evaluadas. Posee *plugins* que permiten acceder a los datos y características de los dispositivos, además de no generar efectos monetarios de desarrollo debido a sus términos de licencia de código abierto. La escalabilidad y mantenimiento van de la mano con éste *framework*, por su constante actualización ante la demanda del mercado.

IV.3.2 Selección de tecnología para el almacenamiento de datos, notificaciones *push* y gestión de usuarios

El comportamiento correcto del aplicativo móvil depende del almacenamiento de la información adecuada y coherente de los datos, por ello, es importante la herramienta a escoger para la gestión, en éste caso, una tecnología que satisfaga las necesidades de almacenamiento de datos en línea o en la nube (*cloud storage*).

Existen numerosas plataformas que brindan servicios de *cloud storage* para desarrolladores de aplicaciones móviles, donde figuran Google Firebase y Amazon Web Services (a través de Amazon DynamoDB) como herramientas adaptadas a las necesidades del aplicativo móvil.

También es importante para el desarrollo de todas las funcionalidades y funcionamiento eficaz del aplicativo móvil, el empleo de una plataforma que permita el manejo de notificaciones *push*. Las plataformas a evaluar para el empleo de esta funcionalidad son Google Firebase y Amazon Web Services (a través de Amazon SNS).

El análisis para la selección de la plataforma que permite el almacenamiento de los datos del aplicativo móvil y la manipulación de notificaciones push se llevó a cabo tomando en cuenta las dos alternativas descritas anteriormente, verificando que cumplan con los requerimientos y especificaciones necesarias a ser utilizadas, asignando dos puntos a la plataforma que cumpla completamente con la característica evaluada, uno si cumple a medias y cero si no cumple.

La selección de la plataforma adecuada para se basa en las siguientes características:

- Adaptabilidad: Capacidad de adaptarse a cualquier entorno de desarrollo (lenguajes de programación, *frameworks*, entre otros).
- Costo: Valor monetario que supone el desarrollo del aplicativo móvil con el uso de la plataforma evaluada.
- Mantenimiento: Característica que posee un sistema para permitir a administradores realizar tareas para mantener su funcionamiento eficaz. Es evaluado el esfuerzo empleado en el mantenimiento de la plataforma.

En la Tabla 3. – Comparación entre plataformas de *cloud computing*, muestra el resultado obtenido de la evaluación de las plataformas de *cloud computing* según las características modelos a seguir.

Tabla 3. – Comparación entre plataformas de *cloud computing* [Elaboración propia]

	Google Firebase	Amazon Web Services
Adaptabilidad	2	1
Costo	2	2
Mantenimiento	2	2
Total	6	5

Tanto Firebase como AWS presentan una fácil adaptación con aplicaciones *web*, sin embargo, el primero se acopla de una manera más eficiente con el *framework* de desarrollo híbrido seleccionado (Apache Cordova), mediante el empleo

del *framework* de desarrollo *web* basado en JavaScript AngularJS, a través de AngularFire.

Por lo descrito, la herramienta fue Google Firebase, cumpliendo con las necesidades del aplicativo móvil; funcionando como fuente de datos en tiempo real y herramienta para gestión de notificaciones *push*, mejorando la experiencia del usuario.

IV.3.3 Selección de tecnologías para el desarrollo del motor de búsqueda

Dada la necesidad de crear una aplicación funcional que permita recolectar las prendas de vestir publicadas en la tienda virtual de **Amazon.com**, se considera la búsqueda de un lenguaje de programación robusto y documentado, además de un sistema de base de datos relacional que, trabajando en conjunto, extraigan los datos requeridos (datos de las prendas de vestir y marcas relacionadas) de las páginas webs, mediante el empleo de la técnica de *Web Scraping*.

En el mercado de desarrollo de aplicaciones existen numerosos lenguajes de programación de código abierto. Por su parte, Java lidera actualmente dicho mercado, según TIOBE. [43]

Java es un rápido, seguro y fiable lenguaje de programación originalmente desarrollado por James Gosling y comercializado por primera vez en 1995 por Sun Microsystems. Es un lenguaje multiplataforma orientado a objetos, que emplea una máquina virtual propia (Java Virtual Machine) para la generación de los compilados.

Java se adapta fácilmente a las necesidades del motor de búsqueda que se desea desarrollar, a través de la implementación de librerías y componentes que permiten emplear la técnica de *Web Scraping*.

Selenium es un entorno de pruebas de software para aplicaciones basadas en la web. Provee una herramienta para grabar/reproducir pruebas con el uso de un amplio número de lenguajes de programación, incluido entre ellos Java, permitiendo

ejecutar dichas pruebas en la mayoría de los navegadores web en distintos sistemas operativos como Windows, Linux y Mac OS.

Por lo descrito, el lenguaje seleccionado para el desarrollo del motor de búsqueda (web crawler) de prendas de vestir fue Java, debido a su implementación *Open JDK*, versión libre de la plataforma de desarrollo Java, su característica multiplataforma y su acoplamiento con el componente Selenium para el empleo de *Web Scraping*.

Por otra parte, existen numerosos sistemas de almacenamiento de datos relacional que lideran el mercado actual. Dada la necesidad que presenta éste proyecto de almacenar de forma segura y persistente las prendas de vestir extraídas de la *web* por el motor de búsqueda, se considera la selección de un gestor de base de datos de código abierto que permita un almacenamiento sin costos monetarios.

Las alternativas a contemplar para la selección del sistema de gestión de bases de datos relacionales son MySQL, Oracle y SQL Server, siendo éstos los más usados actualmente en el mercado.

El análisis para la selección del sistema de gestión de bases de datos que permite el almacenamiento de los datos extraídos por el motor de búsqueda se llevó a cabo tomando en cuenta las tres alternativas nombradas anteriormente, verificando que cumplan con los requerimientos y especificaciones necesarias a ser utilizadas, asignando un puntos al gestor que cumpla con la característica evaluada y cero en caso contrario.

La selección del gestor de bases de datos adecuado para se basa en las siguientes características:

- Costo: Valor monetario que supone la implementación del gestor de bases de datos. Se requiere que el sistema de gestión seleccionado no implique costo en la implementación.

- Adaptabilidad: Capacidad que posee el sistema para adaptarse a cualquier entorno de desarrollo (lenguajes de programación).
- Mantenimiento: Característica que posee un sistema para permitir a administradores realizar tareas para mantener su funcionamiento eficaz. Es evaluado el esfuerzo empleado en el mantenimiento de la base de datos.
- Persistencia: Capacidad de mantener persistentes los datos en caso de fallas.

En la Tabla 4. – Comparación entre sistemas de gestión de bases de datos, muestra el resultado obtenido de la evaluación de los gestores según las características modelos a seguir.

Tabla 4. – Comparación entre sistemas de gestión de bases de datos [Elaboración propia]

	MySQL	Oracle	SQL Server
Costo	1	0	0
Adaptabilidad	1	1	1
Mantenimiento	1	1	1
Persistencia	1	1	1
Total	4	3	3

MySQL es un sistema de gestión de bases de datos de código abierto, mientras que Oracle y SQL Server poseen un alto costo de implementación.

Por lo descrito, MySQL fue la base de datos seleccionada para soportar la información extraída de la ejecución del motor de búsqueda (Web crawler).

IV.4 SPRINTS

Siguiendo los estándares expuestos por el marco de trabajo utilizado (Scrum), se elaboró el *product backlog* y los *sprint backlogs* utilizados durante cada iteración; el detalle de estos se encuentra descrito en el Apéndice A.

El desarrollo del sistema se desglosa de la siguiente manera, describiendo el proceso de cada iteración:

IV.4.1 Sprint 1 – Base de datos, Módulo de Registro y Módulo de Configuración inicial

La presente fase se encuentra dividida en dos partes: diseño e implementación de la estructura de almacenamiento de datos apto para la captura y manipulación de información del sistema finalizando con el registro del usuario y la configuración inicial del perfil general, permitiendo el acceso y uso de la aplicación de manera adecuada.

En el Apéndice A, la Tabla 8. – Sprint Backlog 001 y la Tabla 9. – Sprint Backlog 002 describen los *sprint backlogs* involucrados en esta iteración.

IV.4.1.1 Módulo de Base de datos

La base de datos del aplicativo móvil requirió un diseño de un modelo de datos que soporte la información que maneja cada una de las funcionalidades desarrolladas, permitiendo un comportamiento pertinente del sistema. El diseño engloba la información referente a marcas de prendas de vestir, tallas, tipos de prendas de vestir, áreas de medida (partes del cuerpo), usuarios y mensajes. Como la tecnología elegida para el almacenamiento de los datos fue Google Firebase, por normativas de esta herramienta, el formato utilizado fue JSON conformado en una relación llave/referencia y su valor respectivo.

El almacenamiento de los datos principales del sistema se aloja en la nube (Firebase). En una base de datos local dentro del dispositivo móvil, por medio de un almacenamiento de datos web SQLite[14], se aloja la información referente a la configuración inicial del aplicativo móvil.

La estructura de la base de datos de Google Firebase está conformada por siete nodos padres. A continuación, se establece una breve descripción de cada uno de ellos:

[14] **SQLite**: es un sistema de gestión de base de datos relacional, normalmente utilizado en aplicaciones web.

- *brands*: corresponde a las marcas mostradas por el aplicativo móvil.
- *featured-brands*: almacena las marcas más populares.
- *frames*: consta de todas las tallas manejadas por el aplicativo, divididas por sexo para una mejor distribución.
- *items*: nodo que contiene todos los tipos de prendas de vestir que gestiona el aplicativo móvil.
- *measureAreas*: nodo que posee las partes del cuerpo mostradas en el aplicativo móvil.
- *sms*: nodo que contiene los mensajes de texto con los códigos que no han sido verificados.
- *users*: nodo que posee toda la información de los usuarios registrados en el aplicativo móvil, posee información correspondiente a perfiles, notificaciones, datos personales y configuraciones.

En la Figura 6. – Modelo de datos NoSQL (users), se muestra el formato en el cual se almacenan los datos del nodo *users* en Google Firebase.

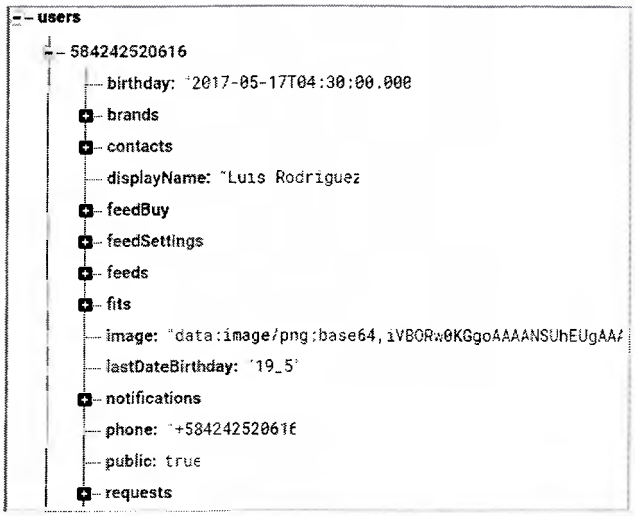

Figura 6. – Modelo de datos NoSQL (users) [Elaboración propia]

En el Apéndice B, la Figura 7. – Modelo de base de datos NoSQL 1 y la Figura 8. – Modelo de base de datos NoSQL 2 muestran la estructura completa

de los nodos que conforman el modelo de datos manejados en la plataforma Google Firebase para el almacenamiento de data en tiempo real.

IV.4.1.2 Desarrollo del módulo de Registro

Según los criterios del sistema, se debe de tener registro de los usuarios dentro del sistema, con dicha finalidad, es de necesidad un módulo que registre al usuario dentro del aplicativo móvil y llevar seguimiento de sus preferencias y acciones permitidas.

El registro del usuario se bifurca en dos alternativas, la primera opción, a través de un código de autorización por mensajería de texto y la segunda opción, por medio de la autenticación con una cuenta válida de la red social de Facebook.

Registro mediante código de autorización

Registro de un usuario mediante la generación de un código de autorización, enviado por mensajería de texto por medio del número de teléfono registrado al iniciar el aplicativo. Una vez verificado el código de autenticación, es finalizado el proceso satisfactorio del registro del usuario permitiendo el uso de los beneficios provistos por el sistema.

Para el envío del mensaje de texto se empleó una interfaz de programación de aplicaciones (API) ofrecido por plataforma de comunicación en la nube Twilio[15].

Registro mediante autenticación con Facebook

La otra alternativa del registro del usuario es mediante una autenticación con una cuenta válida Facebook, autenticando mediante la utilización de la extensión (*plugin*) de Facebook que provee Apache Cordova.

[15] **Twilio**: Es una empresa de comunicación en la nube que permite a los usuarios usar lenguajes web estándar para crear aplicaciones de voz, VoIP y SMS a través de una API web. [30]

IV.4.1.3 Desarrollo del módulo de Configuración inicial

Este módulo se encarga de registrar los datos principales del usuario (nombre, sexo y talla de su camisa favorita), necesarios para utilizar todas las funcionalidades del aplicativo móvil.

La información del perfil de los usuarios son registrados en la base de datos alojada en la nube (Firebase) para ser accedidos de manera centralizada; mientras que dentro de la base de datos local del dispositivo es almacenada la información que indica culminación del registro y configuración personal del usuario.

Luego de finalizar la configuración inicial, se registra en la tabla **config** de la base de datos local del aplicativo los datos necesarios para indicar que el usuario ha culminado su registro y configuración inicial. Los datos registrados son: permitir peticiones de tallas (*true*), permitir peticiones de conexión (*true*), si el registro está completado (*true*), número de teléfono y unidad de medida por defecto (in).

En el Apéndice B, la Figura 9. – Proceso de registro y configuración inicial, muestra los pasos que sigue un usuario cuando ejecuta por primera vez el aplicativo móvil, y la Figura 10. – Tabla config de la base de datos local, muestra el estado de la tabla de configuración luego de haber realizado la configuración inicial.

IV.4.2 Sprint 2 – Módulo de Gestión de tallas y prendas

Se desarrollaron todos los procesos involucrados en el manejo de las tallas y las prendas de vestir de la aplicación. Este sprint se encuentra dividido en tres partes:

- Investigación de los datos necesarios sobre las prendas de vestir, tallas y medidas de las partes del cuerpo que manejará el aplicativo móvil.
- Procesos relacionados con la gestión de las prendas de vestir.
- Procesos relacionados con el manejo de las tallas y medidas del cuerpo.

En el Apéndice A, la Tabla 10. – Sprint Backlog 003 describe el *sprint backlog* involucrado en esta iteración.

A continuación, se desglosan las fases presentadas en la iteración:

IV.4.2.1 Investigación de prendas, tallas y medidas de partes del cuerpo

Durante la investigación, se analizaron las prendas de vestir manejadas por Amazon.com, dado que serán utilizados y gestionados por el aplicativo móvil.

En la Tabla 5. – Prendas de vestir y en la Tabla 6. – Tipos de prendas de vestir se muestran las prendas de vestir y sus tipos, con las que el usuario y el aplicativo móvil pueden interactuar.

Tabla 5. – Prendas de vestir [Elaboración propia]

Prenda de vestir
Blazers
Bras
Capris
Coats
Dresses
Jackets
Jeans
Leggins
Norman
Pants
Shirts
Shoes
Swimwear
Tshirts

Tabla 6. – Tipos de prenda de vestir [Elaboración propia]

Tipos de prenda de vestir
Loose
Regular
Relax
Skinny
Slim

También se investigó qué partes del cuerpo y medidas equivalen a las tallas de cada prenda de vestir.

En la Tabla 17. – Tallas y medidas (Hombre) y la Tabla 18. – Tallas y medidas (Mujer) ubicadas en el Apéndice B, se puede observar la descripción de cada una de las tallas manejadas por el aplicativo, así como también las partes del cuerpo que influyen en la medición de las prendas de vestir, junto con sus respectivas medidas.

Sin importar qué prenda de vestir se desea analizar, las medidas de las partes del cuerpo siempre van a ser las mostradas en las dos tablas mencionadas anteriormente.

En la Tabla 19. – Prendas y partes del cuerpo, ubicada en el Apéndice B, se muestran cuales partes del cuerpo influye en la medición de cada prenda de vestir.

IV.4.2.2 Gestión de prendas de vestir

Los usuarios tienen la opción de gestionar sus prendas, agregándolas y modificándolas en sus favoritas. Por ello, el módulo permite gestionar las prendas deseadas del usuario.

En este proceso, el usuario selecciona la prenda de vestir seguido del tipo de prenda que se encuentre dentro de las opciones listadas. También se implementó la opción de modificar alguna de las prendas agregadas a la lista de favoritas, seleccionando otro tipo de prenda.

En el Apéndice B, en la Figura 11. – Proceso de adición de prenda favorita se muestran los pasos que sigue un usuario para agregar una prenda de vestir a la lista de favoritas, y en la Figura 12. – Proceso de modificación de prenda favorita se muestran los pasos para modificar una prenda de vestir agregada en la lista de favoritas.

IV.4.2.3 Gestión de tallas

El sistema permite al usuario configurar las medidas del cuerpo como características de selección de tallas.

En el ingreso de las medidas del cuerpo, el aplicativo móvil presenta una lista de selección de partes del cuerpo con el motivo de establecer las medidas correspondientes a cada parte, figurado en pulgadas o centímetros (in/cm) según el usuario decida; medidas que poseen la disponibilidad para su modificación posterior.

Adicionalmente para ambas funcionalidades, el aplicativo móvil permite al usuario reproducir un video previo al establecimiento de las medidas de las partes del cuerpo, en donde se describen los pasos que se deben seguir para medir de forma correcta la parte del cuerpo seleccionada.

En el Apéndice B, en la Figura 13. – Proceso de adición de medida de parte del cuerpo se muestran los pasos que sigue un usuario para agregar una medida de una parte del cuerpo, y en la Figura 14. – Proceso de modificación de medida de parte del cuerpo se muestran los pasos para modificar una medida de una parte del cuerpo agregada previamente.

IV.4.3 Sprint 3 – Módulo de Gestión de marcas y Módulo de Gestión de perfiles

Se desarrollaron los todos los procesos involucrados en el manejo de las marcas de prendas de vestir y la gestión de los perfiles de usuarios. Este *sprint* se encuentra dividido en dos partes, la primera consta del desarrollo de las funcionalidades relacionadas con el manejo de las marcas, y la segunda engloba el desarrollo de las funcionalidades relacionadas con los perfiles de usuarios del aplicativo móvil.

En el Apéndice A, la Tabla 11. – Sprint Backlog 004 y la Tabla 12. – Sprint Backlog 005 describen los *sprint backlogs* involucrados en esta iteración.

IV.4.3.1 Gestión de marcas

Éste módulo permite al usuario establecer sus marcas de prendas de vestir preferidas. El aplicativo móvil muestra una lista de selección de las marcas gestionadas, permitiendo al usuario seleccionar una marca como favorita o desmarcar una previamente establecida como favorita. Adicionalmente, el aplicativo posee una vista que muestra las marcas más destacadas.

En el Apéndice B, en la Figura 15. – Proceso de visualización y selección de marcas favoritas se muestran los pasos que se deben seguir los usuarios para ver las marcas destacadas y seleccionar sus marcas favoritas.

IV.4.3.2 Gestión de perfiles

Los perfiles están compuestos por cada uno de los datos que agregan los usuarios, es decir, medidas de las partes del cuerpo, tipos de prendas de vestir favoritas, marcas favoritas y prendas almacenadas obtenidas de las búsquedas en Amazon.com. Por lo tanto, los datos que manejan los perfiles son almacenados en los objetos de tipo users en la base de datos de Google Firebase.

Luego de que un usuario realiza la configuración inicial, se crea el perfil. Al modificar o agregar algún dato, como por ejemplo, un tipo de prenda de vestir favorita o una dirección de correo, automáticamente el perfil será actualizado.

El módulo permite al usuario gestionar su perfil dentro del aplicativo móvil, manipulando el nombre del usuario, la fecha de cumpleaños, el número de teléfono, el correo electrónico y el sexo; y adicionalmente, agregar o modificar tipos de prendas de vestir favoritas, medidas de partes del cuerpo y marcas favoritas

IV.4.4 Sprint 4 – Módulo de Gestión de contactos y Gestión de notificaciones

Se desarrollaron todos los procesos involucrados en la gestión de contactos y notificaciones. La iteración fue dividida en dos partes, una dedicada el desarrollo de las funcionalidades relacionadas con la gestión de contactos y otra dedicada al desarrollo de las funcionalidades relacionadas con la gestión de las notificaciones recibidas por un usuario.

En el Apéndice A, la Tabla 13. – Sprint Backlog 006 y la Tabla 14. – Sprint Backlog 007 describen los *sprint backlogs* involucrados en esta iteración.

IV.4.4.1 Gestión de contactos

La gestión de los contactos de los usuarios permite generar solicitudes de conexión desde un usuario a otro almacenado como contacto en la agenda del dispositivo en uso. El proceso de despliegue de la lista de selección del contacto de la agenda telefónica del dispositivo es realizado por medio de la herramienta *cordova-plugins-contacts* propiciada por el *framework* Apache Cordova.

Se implementó una funcionalidad que permite a los usuarios ver el perfil de sus contactos, siempre y cuando no se encuentren privados. Para esto, los usuarios deben ir a la sección de contactos y seleccionar el contacto el cual desean ver su perfil, donde se pueden observar las prendas preferidas, medidas de partes del cuerpo, marcas favoritas y búsquedas publicadas personales de cada uno.

Adicionalmente, un usuario puede solicitar a un contacto en específico, la medida de alguna parte del cuerpo; para ello, el usuario debe ingresar a la sección de contactos y seleccionar la opción de solicitar medida en el contacto que desee.

En el Apéndice B, en la Figura 16. – Proceso de solicitud de conexión con contacto se muestran los pasos para enviar una solicitud a una persona

que se encuentre en la agenda del teléfono donde se ejecuta el aplicativo, y en la Figura 17. – Solicitud de medida de parte del cuerpo se muestra el proceso para realizar una solicitud de medida de parte del cuerpo de un contacto.

Las notificaciones de conexión entre contactos se describen en la sección de Gestión de notificaciones.

IV.4.4.2 Gestión de notificaciones

Las notificaciones son esenciales en el aplicativo móvil. Brindan una interacción dinámica con el usuario y lo incentiva a usar el aplicativo, además de recibir notificaciones de noticias o alertas de actividades preferenciales.

Un usuario puede recibir notificaciones cuando un contacto: está cumpliendo años, solicita una medida, envía una conexión, aprueba una solicitud, selecciona me gusta (*like*) a una prenda de vestir registrada por el usuario o actualiza su perfil. También un usuario recibe notificaciones cuando obtiene un *match* perfecto al realizar una búsqueda. Las notificaciones son presentadas en tiempo real, enviadas a los usuarios a través de la herramienta de gestión de notificaciones push que provee Firebase y se actualizan automáticamente en el aplicativo móvil gracias a la tecnología de AngularJS, donde permite ejecutar procesos al momento de percibir cambios entre el monitoreo de sus variables.

Los tipos de notificaciones del aplicativo móvil se desglosan a continuación:

- Las alertas de los cumpleaños de los contactos son manejadas realizando comparaciones entre la fecha actual y la fecha de los cumpleaños de los contactos agregados, donde cada coincidencia genera una notificación al usuario que está ejecutando el aplicativo.
- Las notificaciones de enlace entre usuarios constan de lo siguiente: el usuario emisor envía la solicitud y el usuario receptor tiene la

posibilidad de aceptar la solicitud de conexión del emisor o en su defecto, rechazarla.

- Las notificaciones de solicitud de medida de parte del cuerpo se gestionan de la siguiente manera: si el usuario ya agregó a su perfil la medida solicitada por el contacto anteriormente, éste puede seleccionarla y enviarla, en caso contrario, puede agregar la medida y enviarla al contacto que la solicitó. También puede rechazar ésta solicitud.
- Las notificaciones de *match* perfecto se generan cuando un usuario luego de realizar una búsqueda de prendas de vestir, obtiene un *match* perfecto en alguna de las prendas conseguidas. El usuario recibirá una notificación por cada *match* perfecto.
- Las notificaciones de gustar (*like*) a una prenda se generan cuando un contacto de un usuario opta por visitar su perfil y ver las prendas que el usuario ha registrado (seleccionando la opción de *like*) luego de los procesos de búsqueda. El contacto al seleccionar gustar (*like*) generará la notificación al usuario.
- Las notificaciones de actualización de perfil se generan cuando un contacto modifica alguno de sus datos personales (nombre, fecha de cumpleaños, número de teléfono, correo e imagen de perfil). El usuario recibirá la notificación cuando el contacto guarde los cambios realizados.

En el Apéndice B, la Figura 18. – Visualización de notificaciones se muestra cómo se observan las notificaciones de un usuario en el aplicativo móvil.

IV.4.5 Sprint 5 – Módulo de búsqueda, comparación y emparejamiento

Una vez configuradas las premisas de búsqueda de las prendas de vestir, es necesario un módulo o funcionalidad que permita realizar las búsquedas especializadas. El módulo consta de dos procesos relevantes:

- Recolección y almacenamiento de prendas de vestir disponibles a la venta en Amazon.com.
- Consulta de prendas de vestir especializadas para el usuario por medio del aplicativo móvil.

Estas funcionalidades engloban el epicentro del aplicativo al permitir la búsqueda y selección de prendas bajo parámetros volátiles para cada ejecutor.

En el Apéndice A, la Tabla 15. – Sprint Backlog 008 describe el *sprint backlog* involucrado en esta iteración.

IV.4.5.1 Recolección y almacenamiento de prendas de vestir

La recolección de datos es realizada por medio de un motor de búsqueda en la web (*web crawler*) sobre elementos específicos (prendas de vestir) para luego ser almacenados en una base de datos relacional mediante el envío de información por servicios *web restful*.

El *web crawler* se encarga de realizar las búsquedas sobre la tienda virtual de Amazon.com, obteniendo las prendas de vestir disponibles al público, bajo parámetros de selección por sexo y tipos de prendas de vestir relacionadas.

Al obtener los detalles de los productos especificados dentro de la búsqueda, éstos son almacenados en la base de datos para su futuro uso. La instrucción de almacenamiento de los datos capturados procede al llamado del servicio *web* de grabado de datos para su registro persistente.

Las consultas de los datos recolectados se realizan igualmente a través de un servicio *web* de consulta de productos bajo los parámetros de distinción cumpliendo con los valores del tipo de prenda, talla y sexo solicitados. Dicho servicio es utilizado por la aplicación móvil para obtener las prendas de vestir acorde a las características predefinidas por el usuario interesado.

IV.4.5.2 Consulta prendas de vestir a través del aplicativo móvil

Las consultas de las prendas dentro del aplicativo móvil son filtradas y calculadas en base a las medidas de las partes del cuerpo previamente configuradas por el usuario, seguido de la evaluación de los tipos de prendas de vestir que se componen de las partes del cuerpo que el usuario ha ingresado en el aplicativo. Dichos cálculos permiten que el aplicativo proceda a realizar la solicitud de prendas de vestir requeridas a través de peticiones arrojadas al servicio *web* especializado.

Una vez obtenido el resultado final, se procede a verificar los rangos de precios y porcentajes de exactitud establecidos por el usuario.

Cada prenda que cumpla con los rangos, el aplicativo la muestra indicando imagen principal, nombre de la prenda, precio, imagen y nombre de la marca y porcentaje de exactitud. Adicionalmente, el aplicativo permite realizar dos (2) operaciones, una permite dar gustar (*like*) a la prenda donde automáticamente genera la instrucción de almacenamiento con fines de registros para consultas posteriores, y otra permite dar comprar, donde se establece un enlace de re-direccionamiento a la página de la prenda publicada en la tienda virtual de **Amazon.com**.

Se pueden detallar los siguientes puntos de las prendas que hayan cumplido con los requerimientos de búsqueda al seleccionarlas: imagen(es), precio, nombre, detalle de porcentaje de exactitud.

En el Apéndice B, Figura 20. – Consulta de prendas se muestra el proceso de consulta, visualización en detalle y selección de opciones de gustar (*like*) y compra sobre las prendas de vestir consultadas.

IV.4.6 Sprint 6 – Módulo de Configuraciones y ajustes

Este sprint estuvo dedicado a la implementación de las funcionalidades que permiten al usuario establecer las configuraciones del aplicativo móvil.

En el Apéndice A, la Tabla 16. – Sprint Backlog 009 describen el *sprint backlog* involucrado en esta iteración.

Las configuraciones se encuentran divididas en cuatro partes, como se lista a continuación:

- Selección de la unidad de medida por defecto de las partes del cuerpo: en donde el usuario posee dos alternativas para seleccionar, pulgadas (in) y centímetros (cm). Cabe destacar, que al momento de agregar o modificar una medida de alguna parte del cuerpo, el usuario puede seleccionar con que unidad de medida desea hacerlo.
- Elección de privacidad de perfil: en donde el aplicativo permite al usuario establecer su perfil como público o privado.
- Elección de rango de precios y porcentaje de exactitud: donde el aplicativo ofrece al usuario un rango de precios y porcentajes por defecto, permitiendo modificarlos según las necesidades o deseos del usuario. Esta elección guarda una estrecha relación con las búsquedas y comparaciones.
- Elección de notificaciones: donde el aplicativo permite que el usuario seleccione qué notificaciones desea recibir de una lista mostrada en el aplicativo móvil.

En el Apéndice B, Figura 21. – Configuraciones en el aplicativo móvil, se observa cómo un usuario puede manipular las configuraciones.

CAPÍTULO V – RESULTADOS

Una vez culminado el desarrollo del proyecto, incluyendo sus pruebas funcionales, se puede decir que todos los objetivos planteados en este trabajo instrumental de grado fueron cumplidos con éxito, cumpliendo con la planificación estipulada y obteniendo los resultados esperados.

Diseñar e implementar un modelo de base de datos para que contemple la información requerida por la aplicación

Se obtuvo un modelo de datos para el manejo de los datos utilizados por el aplicativo móvil, con el objetivo de dar soporte a todas las funcionalidades desarrolladas. Dicho modelo de datos se implementó en la plataforma de base de datos en tiempo real Google Firebase, lo que permite que el tiempo de respuesta sea corto, aumentando la rapidez del aplicativo y mejorando la experiencia de los usuarios finales.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 1.

Desarrollar un módulo de Registro

Se obtuvo el módulo de registro, en donde el usuario final puede realizar su ingreso al aplicativo móvil de dos formas. La primera, a través de verificación de teléfono, donde el usuario ingresa su número de teléfono y obtiene un mensaje con un código de verificación. La segunda consta de una funcionalidad que permite al usuario final registrarse mediante el ingreso de sus credenciales de Facebook.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 1.

Desarrollar un módulo de Configuración inicial

Se obtuvo el módulo de configuración inicial, en donde el usuario debe ingresar su nombre, sexo y la talla de su camisa favorita. Si el usuario previamente

realizó su registro mediante Facebook, automáticamente el aplicativo solicita únicamente la talla de su camisa favorita, ya que el nombre y sexo se extraen de la cuenta de Facebook ingresada por el usuario.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 1.

Desarrollar un módulo de Gestión de tallas y prendas

Se obtuvo el módulo de gestión de tallas y prendas, el cual está divido en dos sub módulos. En el primer sub módulo, dedicado a las tallas y medidas de partes del cuerpo, el usuario tiene la posibilidad de agregar o modificar sus medidas de las partes del cuerpo que el aplicativo móvil maneja. Adicionalmente, en el momento de realizar cualquiera de las dos opciones mencionadas anteriormente, el usuario tiene a disposición la reproducción un video que explica la forma correcta de medir la parte del cuerpo seleccionada. En el segundo sub módulo, dedicado a las prendas de vestir, el usuario tiene la posibilidad de agregar o modificar sus tipos de prendas de vestir favoritas, elegidas de la lista de prendas de vestir que el aplicativo maneja.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 2.

Desarrollar un módulo de Gestión de marcas

Se obtuvo el módulo de gestión de marcas, en donde el usuario puede seleccionar o deseleccionar sus marcas favoritas de la lista de marcas manejadas por el aplicativo móvil.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 3.

Desarrollar un módulo de Gestión de perfiles

Se obtuvo el módulo de gestión de perfiles. Este módulo es el encargado de mostrar los datos y la información del usuario, detallando sus medidas, prendas de vestir y marcas favoritas. Cabe destacar que al modificar alguno de los datos mencionados anteriormente, consecuentemente se está modificando el perfil también.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 3.

Desarrollar un módulo de Gestión de contactos

Se obtuvo el módulo de gestión de contactos. El módulo permite al usuario enviar solicitudes de conexión a otros usuarios del aplicativo. Dichos usuarios son obtenidos de la agenda de contactos del teléfono en donde se esté ejecutando el aplicativo móvil. Adicionalmente, el módulo permite al usuario ver el perfil de sus contactos, siempre y cuando no se encuentren privados. Además, un usuario puede solicitarle la medida de una parte del cuerpo a un contacto.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 4.

Desarrollar un módulo de búsqueda, comparación y emparejamiento de prendas de vestir

Se obtuvo el módulo de búsqueda, comparación y emparejamiento, el cual se encuentra dividido en dos sub módulos. El primero consta del *web crawler*, que permitirá almacenar las prendas de vestir que se encuentran a la venta en Amazon.com en una base de datos relacional. Esta aplicación será ejecutada por los administradores del sistema. El segundo sub módulo consta de la sección del aplicativo móvil dedicada a la búsqueda de las prendas que más se ajustan a las características del usuario. Para ello se debe ingresar en la sección *feed*, en donde el aplicativo móvil automáticamente realizará el cálculo de las tallas en base a las medidas de las partes del cuerpo que el usuario ha agregado y a las prendas de vestir

que engloban dichas partes del cuerpo, tomando en cuenta las configuraciones establecidas por el usuario (rango de precios y porcentajes de exactitud).

Luego de realizar la búsqueda, el usuario tiene dos acciones para realizar por cada prenda de vestir obtenida. La primera es darle gustar (*like*) a la prenda para almacenarla y consultarla nuevamente en un futuro (para consultarla nuevamente debe ingresar en el menú y luego seleccionar *feeds*). La segunda es la opción de compra, en donde el aplicativo direcciona a la página de la prenda en Amazon.com.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 5.

Desarrollar un módulo de Gestión de notificaciones

Se obtuvo el módulo de gestión de notificaciones. Éste módulo permite a los usuarios ser notificados cuando un contacto está cumpliendo años, solicita una talla, envía una conexión, aprueba una solicitud, selecciona me gusta (*like*) a una prenda de vestir registrada por el usuario o actualiza su perfil. También un usuario recibe notificaciones cuando obtiene un *match* perfecto. Las notificaciones son recibidas y mostradas en el teléfono receptor. Además, en el aplicativo, existe una sección dedicada amostrar el listado de las notificaciones recibidas, en donde para cada notificación se puede optar por atenderla o eliminarla; en adición, el aplicativo móvil permite eliminar todas las notificaciones.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 4.

Desarrollar un módulo de Configuraciones y ajustes

Se obtuvo el módulo de configuraciones y ajustes, en donde el usuario puede realizar cuatro acciones principales. La primera, seleccionar la unidad de medida por defecto (cm o in) que se utilizará para las mediciones de las partes del cuerpo. La segunda, hacer público o privado el perfil para la visualización de los contactos. La tercera, elegir el rango de precios y de porcentajes de exactitud para las búsquedas

de prendas de vestir. Y la cuarta, elegir qué tipo de notificaciones recibir de la manejadas por el aplicativo móvil.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 6.

Diseñar los procesos involucrados en la búsqueda de prendas de vestir mediante comparaciones y emparejamientos con el uso de tallas como criterio a través del aplicativo móvil

Los procesos involucrados en la interacción entre el usuario y el aplicativo móvil en el momento de realizar la búsqueda, comparación y emparejamiento, con el fin de obtener como resultado las prendas de vestir que más se ajustas a las características del usuario, involucran la lectura de las medidas de las partes del cuerpo registradas por el usuario, la evaluación de las prendas que emplean dichas partes del cuerpo para la medición de sus tallas y la transformación de las medidas a las tallas adecuadas, para luego consultar los servicios desarrollados para obtener las prendas que cumplan con las especificaciones enviadas a través del mismo.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 5.

Evaluar entre las distintas tecnologías para el desarrollo de plataformas móviles existentes, que brinden mayor escalabilidad y adaptabilidad de las distintas plataformas móviles

Se realizó una investigación detallada de las diferentes alternativas que ofrece el ámbito del desarrollo de aplicaciones móviles, en donde se evaluaron ventajas y desventajas de cada alternativa, ponderando cada una de ellas, para luego poder seleccionar la más apta en cuanto a costos, escalabilidad, mantenimiento y acceso al hardware de dispositivos.

La alternativa seleccionada fue Apache Cordova, *framework* que permitió el desarrollo del aplicativo móvil bajo un lenguaje de programación (JavaScript) y a la vez, la implementación en los sistemas operativos iOS y Android, brindando la mayor escalabilidad y adaptabilidad posible.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en la selección de tecnologías para el desarrollo de aplicaciones móviles híbridas.

Evaluar e implementar interoperabilidad entre las distintas tecnologías para la vinculación de aplicaciones móviles con *Cloud Storage*, notificaciones *push* y gestión de usuarios

Se realizó una investigación detallada de las diferentes alternativas que ofrece el mercado para la vinculación de aplicaciones móviles con plataformas que ofrecen *Cloud Storage*, notificaciones *push* y gestión de usuarios. Las dos alternativas encontradas fueron Google Firebase y Amazon DynamoDB, donde la elegida fue Google Firebase, debido a su robustez y gran popularidad en base a la evaluación de tecnologías realizada. También permite realizar las tres actividades evaluadas (*Cloud Storage*, notificaciones *push* y gestión de usuarios), manteniendo esta cualidad como una gran ventaja, debido a que se hace más eficaz la implementación de dichas actividades a través de una sola plataforma.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, en la ejecución de cada *sprint*.

Implementar interoperabilidad con Amazon.com

A través del *web crawler*, se logró implementar la interoperabilidad con Amazon.com, recorriendo el sitio web y extrayendo los datos requeridos de cada producto conseguido. También, a través del aplicativo móvil se implementó interoperabilidad con Amazon.com, dado que luego de obtener los resultados de las

búsquedas, por cada producto existe la opción de realizar la compra, donde el aplicativo direcciona a la página web del producto correspondiente.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 5.

Implementar interoperabilidad con los servicios de Facebook

A través del aplicativo móvil, se logró implementar interoperabilidad con Facebook en el proceso de registro, en el momento en el que un usuario desea realizar su registro mediante el ingreso de sus credenciales de Facebook, extrayendo, luego de la validación de dichas credenciales, el nombre del usuario y su sexo.

Estos resultados fueron obtenidos en el Capítulo IV – Desarrollo, específicamente en el *sprint* 1.

CAPÍTULO VI - CONCLUSIONES Y RECOMENDACIONES

VI.1 CONCLUSIÓN

En el presente trabajo instrumental de grado se logró obtener un producto que cumple con cada uno de los requerimientos establecidos desde el comienzo de su desarrollo. Además se consiguió que el producto sea ejecutable en los sistemas operativos móviles más usados actualmente en el mercado (iOS y Android).

A través del aplicativo móvil, se facilitará a sus usuarios la búsqueda de prendas de vestir que más se ajusten a sus características físicas (medidas de partes del cuerpo), permitiendo ahorrarse tiempo en la búsqueda de este tipo de productos, además de brindarles la facilidad de obtener el enlace que les permita comprar sus prendas favoritas. También, con la funcionalidad implementada que permite la interacción entre contactos, los usuarios podrán compartir medidas de partes del cuerpo, marcas favoritas y sus prendas consultadas favoritas con sus contactos. También podrán obtener notificaciones generadas de la interacción de los contactos con el aplicativo móvil.

El desarrollo de aplicaciones móviles híbridas posee ventajas importantes durante el desarrollo, siendo las más importantes el corto tiempo de desarrollo y el acoplamiento a distintos sistemas operativos. Pero también posee desventajas, siendo la más significativa el poco control que se posee en cuanto a las acciones directas con el hardware de los dispositivos, debido a que se acceden a ellos a través de *plugins*.

Las aplicaciones móviles requieren de tiempos de respuesta muy cortos para que la experiencia de los usuarios no se vea desmejorada. Los BaaS forman parte de las plataformas que ayudan a que la premisa descrita anteriormente se haga realidad, debido a proveen servicios de bases de datos en tiempo real.

Por otra parte, a través del *web crawler*, los usuarios administradores del sistema podrán actualizar la base de datos de las prendas de vestir, extrayéndolas de

las publicaciones realizadas en Amazon.com. Ésta aplicación será ejecutada en un tiempo periódico establecido por los usuarios administradores. También es necesario el monitoreo de los resultados obtenidos en las ejecuciones del *web crawler*, con la finalidad de verificar que la extracción de datos se realiza correctamente y el sitio web no ha cambiado su estructura.

Durante el desarrollo del *web crawler* se detectó una desventaja que posee la ejecución del proceso de *web scraping*, tornándose complicado en el momento de su ejecución. La desventaja a la cual se hace referencia es la adaptabilidad a cambios en la estructura de las páginas web, ya que se debe reestructurar el código de la aplicación. Sin embargo, percatarse de los cambios no es tarea sencilla, generalmente su descubrimiento viene dado de alguna falla generada en la ejecución del proceso de *web scraping*.

VI.2 RECOMENDACIONES

Este trabajo instrumental de grado no debe verse como solución única para el objetivo por el cual fue desarrollado el producto final. Existen varias formas de mejorar el funcionamiento del aplicativo móvil y el *web crawler*, por ello, a continuación, se listan algunas de ellas:

- Para mejorar y facilitar la experiencia del usuario, se recomienda implementar un módulo de información, encargado de brindar al usuario datos importantes relacionados con las medidas manejadas mundialmente, como por ejemplo, tablas de tallas con sus medidas de partes del cuerpo respectivas o tablas de partes del cuerpo involucradas en la medición de cada prenda de vestir.
- Para expandir la diversidad de prendas de vestir que se obtienen en el proceso de búsqueda del aplicativo móvil, se recomienda desarrollar nuevos *plugins* en el *web crawler* para la extracción de prendas publicadas en otros sitios web proveedores como por ejemplo **eBay**. Para esta recomendación es necesaria la evaluación de los costos que implica en cuanto al mantenimiento de *web crawler*, debido a que con esta expansión el esfuerzo en el monitoreo de la ejecución será mayor.
- Para mejorar y robustecer el registro y configuración inicial en el aplicativo móvil se recomienda la implementación de *login* mediante la red social **Twitter**.
- Para mejorar la comunicación entre contactos se recomienda la implementación de un módulo de conversación que permita intercambiar mensajes directos de contacto a contacto.
- Se recomienda la implementación de una funcionalidad llamada compartir, que permita a los usuarios publicar en sus perfiles las prendas de vestir favoritas de sus contactos.

- Se recomienda el desarrollo de un sistema de monitoreo y administración que permita obtener reportes y estadísticas sobre el comportamiento del motor de búsqueda y la interacción de los usuarios con el aplicativo móvil.
- Se recomienda la implementación de tecnologías de realidad aumentada[16] para realizar las medidas de las partes del cuerpo automáticamente a través de las cámaras de los dispositivos.
- Se recomienda la implementación de un módulo que permita la publicación de prendas de vestir consultadas, en los perfiles de redes sociales de los usuarios.

[16] **Realidad Aumentada**: Agrupa aquellas tecnologías que permiten la superposición, en tiempo real, de imágenes, marcadores o información generados virtualmente, sobre imágenes del mundo físico. [41]

REFERENCIAS BIBLIOGRÁFICAS

[1] Masadelante.com. ¿Qué es un Sistema Operativo?. [En línea]. Disponible: http://www.masadelante.com/faqs/sistema-operativo

[2] Gabriel Pedrozo, Universidad Nacional del Noreste. (2012). Sistemas Operativos en Dispositivos Móviles. [En línea]. Disponible: http://exa.unne.edu.ar/informatica/SO/Sistemas_Operativos_en_Dispositivos_Moviles.pdf

[3] IDC. (Noviembre de 2016). Smartphone OS Market Share, 2016 Q3. [En línea]. Disponible: http://www.idc.com/promo/smartphone-market-share/os

[4] Androidsis. (Marzo de 2014). ¿Qué son las interfaz de usuario de los fabricantes en Android?. [En línea]. Disponible: https://www.androidsis.com/que-son-las-interfaz-de-usuario-de-los-fabricantes-en-android/

[5] Apple, Inc. iOS 10. [En línea]. Disponible: http://www.apple.com/la/ios/ios-10/

[6] Pedro Pimienta, DeIdeaAApp. (Mayo de 2014). Tipos de aplicaciones móviles y sus características. [En línea]. Disponible: https://deideaaapp.org/tipos-de-aplicaciones-moviles-y-sus-caracteristicas/

[7] IBM Corporation Software Group. (Abril de 2012). Desarrollo de aplicaciones móviles nativas, web o híbridas. [En línea]. Disponible: ftp://ftp.software.ibm.com/la/documents/gb/commons/27754_IBM_WP_Native_Web_or_hybrid_2846853.pdf

[8] EcuRed. Lenguaje de programación [En línea]. Disponible: https://www.ecured.cu/Lenguaje_de_Programaci%C3%B3n

[9] Mozila Developer Network. HTML. [En línea]. Disponible: https://developer.mozilla.org/es/docs/Web/HTML

[10] Mozila Developer Network. CSS. [En línea]. Disponible: https://developer.mozilla.org/es/docs/Web/CSS

[11] Mozila Developer Network. JavaScript. [En línea]. Disponible: https://developer.mozilla.org/es/docs/Web/JavaScript

[12] Java Extremo. (Julio de 2012). Características e Historia de JAVA. [En línea]. Disponible: https://javaextremo.wordpress.com/2012/07/31/caracteristicas-e-historia-de-java-9/

[13] CCM. (Marzo de 2017). Introducción a las bases de datos. [En línea]. Disponible: http://es.ccm.net/contents/66-introduccion-a-las-bases-de-datos

[14] IBM. Bases de datos relacionales. [En línea]. Disponible: https://www.ibm.com/support/knowledgecenter/es/SSEPGG_8.2.0/com.ibm.db2.udb.doc/admin/c0004099.htm

[15] Universidad Autonoma del Estado de Morelos. MySQL. [En línea]. Disponible: http://www.gridmorelos.uaem.mx/~mcruz/cursos/miic/MySQL.pdf

[16] IES San Vicente. Que es Oracle. [En línea]. Disponible: https://iessanvicente.com/colaboraciones/oracle.pdf

[17] TechTarget. SQL Server. [En línea]. Disponible: http://searchdatacenter.techtarget.com/es/definicion/SQL-Server

[18] Acens. Bases de datos NoSQL. Qué son y tipos que nos podemos encontrar. [En Línea]. Disponible: https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf

[19] W3C. Guía Breve de Servicios Web. [En línea]. Disponible: http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb

[20] ADWE. (Noviembre de 2013). Servicios web RESTful con HTTP. Parte I: Introducción y bases téoricas. [En línea]. Disponible: http://www.adwe.es/general/colaboraciones/servicios-web-restful-con-http-parte-i-introduccion-y-bases-teoricas

[21] Kin Lane. (Mayo de 2013). Overview Of The Backend as a Service (BaaS) Space. [En línea]. Disponible: https://s3.amazonaws.com/kinlane-productions/whitepapers/API+Evangelist+-+Overview+of+the+Backend+as+a+Service+Space.pdf

[22] José Ángel Zamora. (Mayo 2016). ¿Qué es Firebase? La mejorada plataforma de desarrollo de Google. [En línea]. Disponible: https://www.elandroidelibre.com/2016/05/firebase-plataforma-desarrollo-android-ios-web.html

[23] Google Firebase. (Octubre 2016). Real Time Database. [En línea]. Disponible: https://firebase.google.com/docs/database/

[24] AWS. *Cloud Computing* con Amazon Web Services. [En línea]. Disponible: https://aws.amazon.com/es/what-is-aws/

[25] AWS. Mobile Backend as a Service Patrón de Arquitectura. [En línea]. Disponible: http://aws.typepad.com/espanol/2016/02/mobile-backend-as-a-service-patr%C3%B3n-de-arquitectura.html

[26] AWS. Amazon DynamoDB. [En línea]. Disponible: https://aws.amazon.com/es/dynamodb/

[27] DB-Engines. Amazon DynamoDB System Properties. [En línea]. Disponible: http://db-engines.com/en/system/Amazon+DynamoDB

[28] AWS. DynamoDB Streams and AWS Lambda Triggers. [En línea]. Disponible: http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Streams.Lambda.html

[29] Amazon Web Services. AWS Lambda. [En línea]. Disponible: https://aws.amazon.com/es/lambda/

[30] Amazon Web Services. Amazon Simple Notification Service (SNS). [En línea]. Disponible: https://aws.amazon.com/es/sns/?hp=tile&so-exp=below

[31] Sitelabs. (Abril de 2016). Qué es el Web scraping? Introducción y herramientas. [En línea]. Disponible: https://sitelabs.es/web-scraping-introduccion-y-herramientas/

[32] Proyectosagiles.org. Qué es SCRUM. [En línea]. Displonible: https://proyectosagiles.org/que-es-scrum/

[33] Andrew Phan, Phuong-Van Phan. (2012). *Scrum in Action - Cengage Learning*.

[34] IDS. (02 Diciembre de 2014). *Scrum, 4ta. Parte – Artefactos*. [En línea]. Disponible: http://www.ids.com.mx/desarrollo-profesional/comunidad-ids/blog/scrum-4ta-parte-artefactos

[35] Apache Cordova. Cordova Overview. [En línea]. Disponible: https://cordova.apache.org/docs/en/latest/guide/overview/index.html

[36] J. Alberto. (27 abril 2011). JQueryMobile. [En línea]. Disponible: https://www.genbetadev.com/frameworks/jquerymobile-introduccion-al-desarrollo-web-para-moviles

[37] Luis Setfree. Conociendo Xamarin, la herramienta para desarrolladores multiplataforma. [En línea]. Disponible: http://www.vix.com/es/btg/tech/13263/conociendo-xamarin-la-herramienta-para-desarrolladores-multiplataforma

[38] José Zamora. (Mayo 2014) Xamarin, la API para crear aplicaciones multiplataforma en C#/.NET. [En línea]. Disponible: https://www.elandroidelibre.com/2014/05/xamarin-la-api-para-crear-aplicaciones-multiplataforma-en-c-net.html

[39] Google Firebase. AngularFire. [En línea]. Disponible: https://www.firebase.com/docs/web/libraries/angular/api.html

[40] Android. Android 7.0 Nougat. [En línea]. Disponible: https://www.android.com/versions/nougat-7-0/

[41] Twilio. [En línea]. Disponible: https://www.twilio.com/?lang=null

[42] Blogthinkbig.com. (Agosto de 2012). ¿En qué consiste la realidad aumentada?. [En línea]. Disponible: http://blogthinkbig.com/en-que-consiste-la-realidad-aumentada/

[43] TIOBE. [En línea]. Disponible: https://www.tiobe.com/

APÉNDICE

APÉNDICE A – ARTEFACTOS DE LA METODOLOGÍA

Product Backlog

Tabla 7. – Product Backlog [Elaboración propia]

ID	Historia	Esfuerzo (horas)	Prioridad
AM-001	Como usuario deseo registrarme con el aplicativo móvil	16	8
AM-002	Como usuario deseo ingresar mis datos personales como nombre, fecha de nacimiento, y sexo.	4	8
AM-003	Como usuario deseo agregar medida de mis partes del cuerpo	16	8
AM-004	Como usuario deseo agregar mis tipos de prendas favoritas	16	8
AM-005	Como usuario deseo seleccionar mis marcas de prendas de vestir preferidas	8	8
AM-006	Como usuario deseo agregar contactos	16	8
AM-007	Como usuario deseo configurar que unidad de medida deseo utilizar (in o cm) y qué notificaciones recibir	8	8
AM-008	Como usuario deseo realizar búsquedas de prendas de vestir	80	8
AM-009	Como usuario deseo publicar mi perfil	24	
AM-010	Como usuario deseo comprar una prenda de vestir	24	8
AM-011	Como usuario deseo almacenar una prenda de vestir obtenida de las búsquedas realizadas	16	8

AM-012	Como desarrollador deseo implementar una base de datos local en el aplicativo	4	8
AM-013	Como desarrollador deseo utilizar una base de datos de tiempo real en la nube	4	8
AM-014	Como usuario deseo recibir notificaciones de conexiones con otros usuarios	16	8
AM-016	Como desarrollador necesito implementar llamadas a servicios que retornen las prendas de vestir consultadas en amazon.com	8	8
AM-017	Como desarrollador necesito implementar el api que me permite comunicarme con el servicio proveedor del almacenamiento en la nube	8	8
AM-018	Como desarrollador necesito evaluar que tecnología para el almacenamiento de datos en la nube utilizar	4	8
AM-019	Como desarrollador necesito evaluar que tecnología de desarrollo híbrido usar para el desarrollo del aplicativo móvil	4	8
AM-020	Como desarrollador necesito implementar un web crawler que me permita recorrer amazon.com con el	24	8

	fin de extraer las prendas de vestir a la venta en este sitio web		
AM-021	Como desarrollador necesito implementar una base de datos relacional para el almacenamiento de prendas de vestir extraídas de amazon.com	4	8
AM-022	Como desarrollador necesito implementar servicio webs que permitan almacenar y consultar prendas de vestir	16	8

Sprint Backlog

Tabla 8. – Sprint Backlog 001. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	001
Título	Desarrollo del módulo de registro
Fecha	23/01/2017
Descripción de la historia	Desarrollo de las funcionalidades que abarcan el registro de un usuario en el aplicativo móvil

Tabla 9. – Sprint Backlog 002. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	002
Título	Desarrollo del módulo de configuración inicial
Fecha	23/01/2017

Descripción de la historia	Desarrollo de las funcionalidades necesarias para el flujo de la configuración inicial, es decir, el ingreso de los principales datos del usuario necesarios para el funcionamiento inicial del aplicativo.

Tabla 10. – Sprint Backlog 003. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	003
Título	Desarrollo del módulo de gestión de tallas y prendas
Fecha	08/02/2017
Descripción de la historia	Desarrollo de las funcionalidades necesarias para el manejo de las medidas de las partes del cuerpo y las prendas favoritas.

Tabla 11. – Sprint Backlog 004. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	004
Título	Desarrollo del módulo de gestión de marcas
Fecha	24/02/2017
Descripción de la historia	Desarrollo de las funcionalidades necesarias para el manejo de las marcas favoritas de los usuarios

Tabla 12. – Sprint Backlog 005. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	005
Título	Desarrollo del módulo de gestión de perfiles
Fecha	24/02/2017

Descripción de la historia	Desarrollo de las funcionalidades necesarias para en manejo de los perfiles de los usuarios

Tabla 13. – Sprint Backlog 006. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	006
Título	Desarrollo del módulo de gestión de contactos
Fecha	14/03/2017
Descripción de la historia	Desarrollo de las funcionalidades necesarias para agregar e interactuar con los contactos de un usuario

Tabla 14. – Sprint Backlog 007. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	007
Título	Desarrollo del módulo de gestión de notificaciones
Fecha	14/03/2017
Descripción de la historia	Desarrollo de las funcionalidades necesarias para el manejo de notificaciones recibidas por un usuario

Tabla 15. – Sprint Backlog 008. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	008
Título	Desarrollo del módulo de búsqueda, comparación y emparejamiento
Fecha	30/03/2017

Descripción de la historia	Desarrollo de las funcionalidades necesarias para la consulta de prendas de vestir de amazon.com y la extracción y almacenamiento de prendas de este sitio web.

Tabla 16. – Sprint Backlog 009. [Elaboración propia]

Sprint Backlog	
Aplicativo móvil para consultar y compartir prendas de vestir	
Numero de historia	009
Título	Desarrollo del módulo de configuraciones y ajustes
Fecha	17/04/2017
Descripción de la historia	Desarrollo de las funcionalidades necesarias para el establecimiento de las configuraciones del aplicativo móvil.

APÉNDICE B – ARTEFACTOS DEL DESARROLLO

Sprint 1 – BD, Módulo de Registro y Módulo de Configuración inicial

Figura 7. – Modelo de base de datos NoSQL 1 [Elaboración propia]

Figura 8. – Modelo de base de datos NoSQL 2 [Elaboración propia]

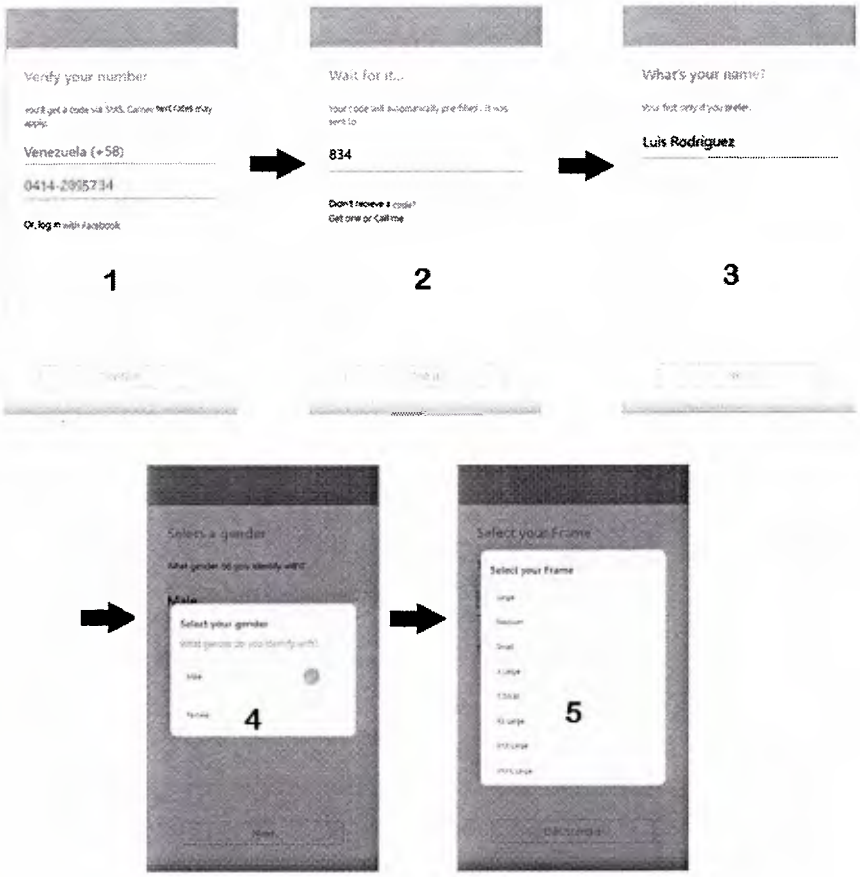

Figura 9. – Proceso de registro y configuración inicial [Elaboración propia]

id	key	value
9	measure	in
10	requestSize	true
11	requestConnection	true
12	registered	true
13	phone	584142895734

Figura 10. – Tabla config de la base de datos local [Elaboración propia]

Sprint 2 – Módulo de Gestión de tallas y prendas

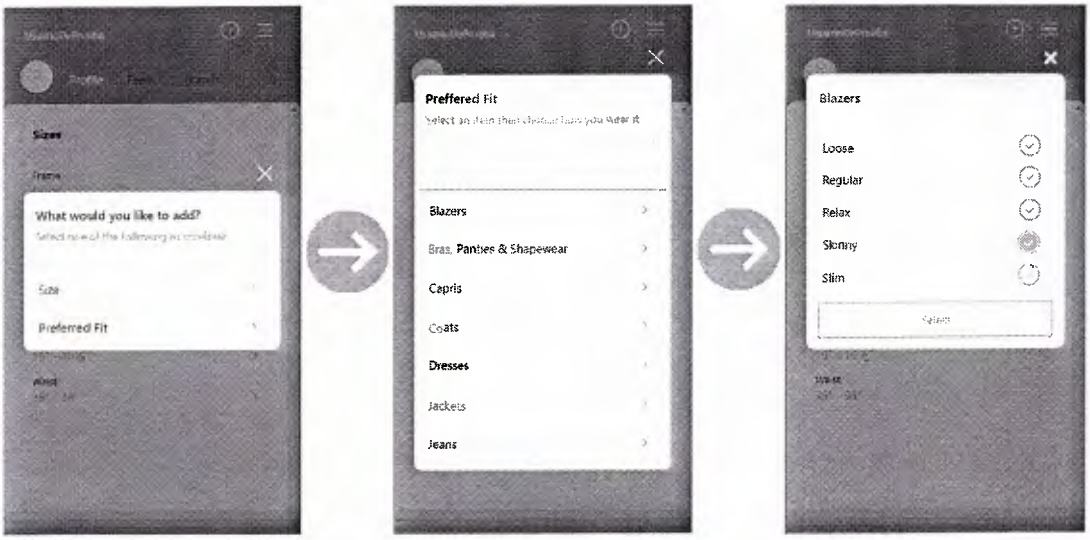

Figura 11. – Proceso de adición de prenda favorita [Elaboración propia]

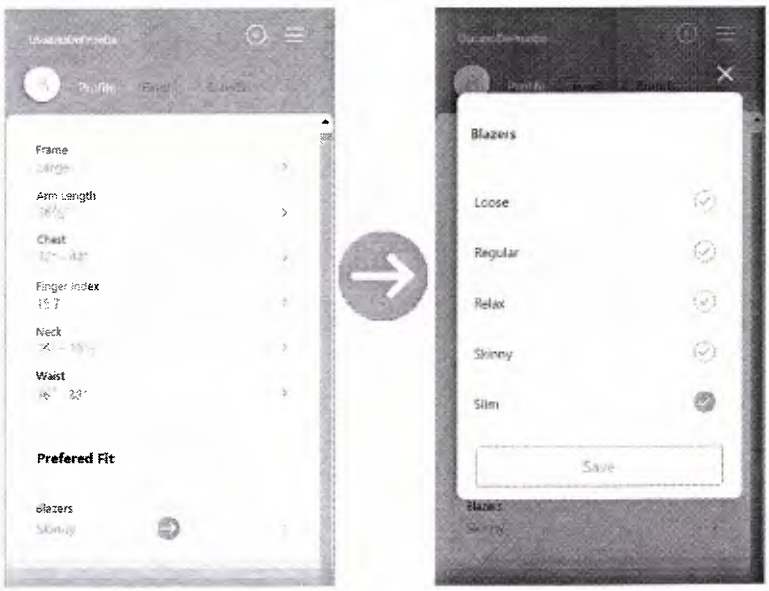

Figura 12. – Proceso de modificación de prenda favorita [Elaboración propia]

Figura 13. – Proceso de adición de medida de parte del cuerpo [Elaboración propia]

Figura 14. – Proceso de modificación de medida de parte del cuerpo [Elaboración propia]

Tabla 17. – Tallas y medidas (Hombre) [Elaboración propia]

		in										
Talla	Código	Largo del brazo	Pecho	Largo del pie	Cabeza	Altura	Cadera	Entrepierna	Cuello	Hombro	Muslo	Cintura
X Small	XS	31.5 – 32	33 – 34	8.5 – 8	6.5 – 6.75	65 – 67	33.25 – 34	29 – 30	13 – 13.5	4.9 – 5.1	18.9 – 19.3	27 – 28
Small	S	32.5 – 33	35 – 37	9 – 9.5	6.875 – 7	67 – 69	35 – 36	30 – 32	14 – 14.5	5.1 – 5.3	19.3 – 19.7	29 – 31
Medium	M	33.5 – 34	38 – 40	9.5 – 10	7.125 – 7.25	69 – 71	37.5 – 38.5	32 – 34	15 – 15.5	5.1 – 5.3	19.7 – 20.1	32 – 34
Large	L	34.5 – 35	42 – 44	10 – 10.5	7.375 – 7.5	71 – 73	39 – 40	34 – 36	16 – 16.5	5.1 – 5.3	20.1 – 20.5	36 – 38
X Large	XL	35.5 – 36	46 – 48	10.5 – 11	7.625 – 7.75	73 – 75	43 – 44.5	36 – 38	17 – 17.5	5.1 – 5.3	20.5 – 20.9	40 – 42
XX Large	XXL	36 – 36.5	50 – 52	11 – 11.5	7.875 – 8	75 – 77	47 – 48.5	38 – 40	18 – 18.5	5.3 – 5.5	20.9 – 21.3	44 – 48
XXX Large	XXXL	36.5 – 37	54 – 56	11.5 – 12	8.125 – 8.25	77 – 78	49 – 50.5	40 – 42	19 – 19.5	5.5 – 5.8	21.3 – 21.7	50 – 52

XXXX Large	XXXXL	37 – 37.5	58 – 60	12 – 12.5	8.375 – 8.5	78 – 79	42 – 44	20 – 20.5	21.7 – 22.1	52 – 54			

Tabla 18. – Tallas y medidas (Mujer) [Elaboración propia]

								in					
Talla	Código	Largo del brazo	Busto	Largo del pie	Cabeza	Altura	Cadera	Entrepierna	Cuello	Hombro	Muslo	Cintura	
X Small	XS	31.5 – 32	31.5 – 32	8.5 – 8	6.5 – 6.75	63.5 – 64	33.25 – 34	29 – 30	13 – 13.5	4.9 – 5.1	17.7 – 18.1	23.25 – 24	
Small	S	32 – 33	33.5 – 34	9 – 9.5	6.875 – 7	64 – 65	35 – 36	30 – 32	14 – 14.5	5.1 – 5.3	18.1 – 18.5	25.5 – 26	
Medium	M	33 – 33.5	35.5 – 36	9.5 – 10	7.125 – 7.25	66 – 67	37.5 – 38.5	32 – 34	15 – 15.5	5.1 – 5.3	18.5 – 18.9	27.5 – 28	
Large	L	33.5 – 34.5	37 – 37.5	10 – 10.5	7.375 – 7.5	67 – 68	39 – 40	34 – 36	16 – 16.5	5.1 – 5.3	18.9 – 19.3	28.5 – 29	
X Large	XL	34.5 – 35	40.5 – 41.5	10.5 – 11	7.625 – 7.75	68 – 68.5	43 – 44.5	36 – 38	17 – 17.5	5.1 – 5.3	19.3 – 19.7	33 – 34.5	
XX Large	XXL	35 – 35.5	44 – 45.5	11 – 11.5	-	68.5 – 69	47 – 48.5	38 – 39	18 – 18.5	5.3 – 5.5	19.7 – 20.1	37 – 38	

Tabla 19. – Prendas y partes del cuerpo [Elaboración propia]

Prenda de vestir	Parte del cuerpo
Blazers	- Largo del brazo - Pecho - Cintura
Bras	- Busto
Capris	- Cintura - Entrepierna
Coats	- Busto - Cadera - Cintura
Dresses	- Busto - Cintura
Jackets	Hombre: - Largo del brazo - Pecho - Cintura Mujer: - Busto - Cadera - Cintura
Jeans	- Cintura - Entrepierna
Leggins	- Cintura - Entrepierna
Norman	- Cintura - Largo del brazo - Hombros
Pants	- Cintura - Entrepierna
Shirts	- Pecho - Cintura
Shoes	- Largo del pie
Shorts	- Cintura
Tshirts	Hombre: - Pecho - Cintura Mujer: - Busto - Cadera - Cintura

Sprint 3 – Módulo de Gestión de marcas y Módulo de Gestión de perfiles

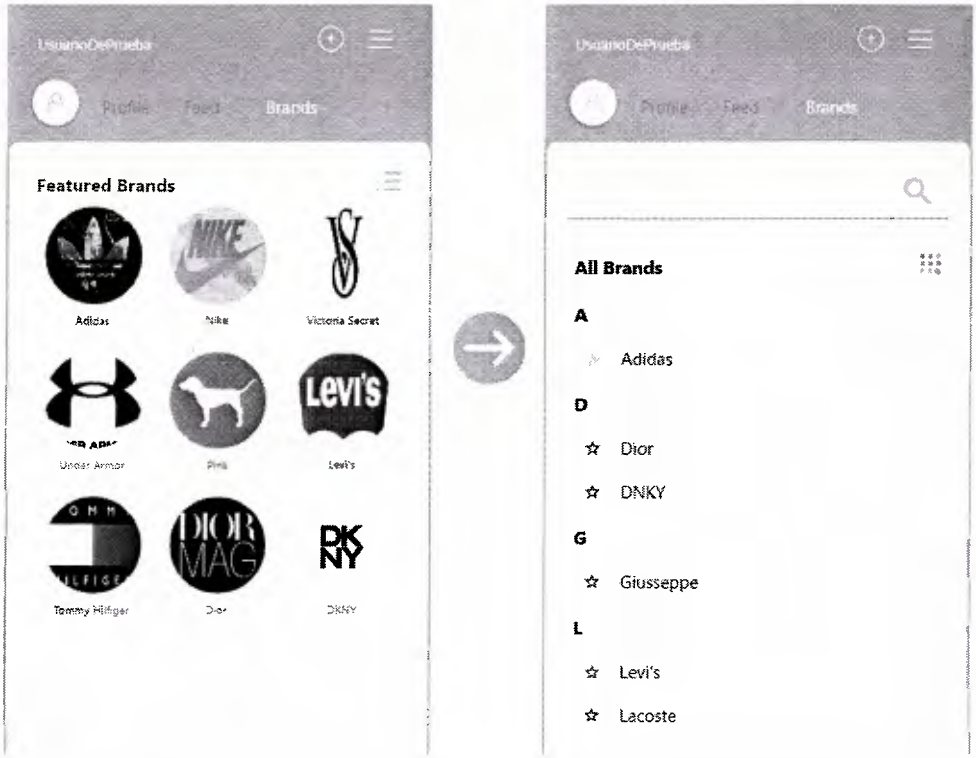

Figura 15. – Proceso de visualización y selección de marcas favoritas [Elaboración propia]

Sprint 4 – Módulo de Gestión de contactos y Gestión de notificaciones

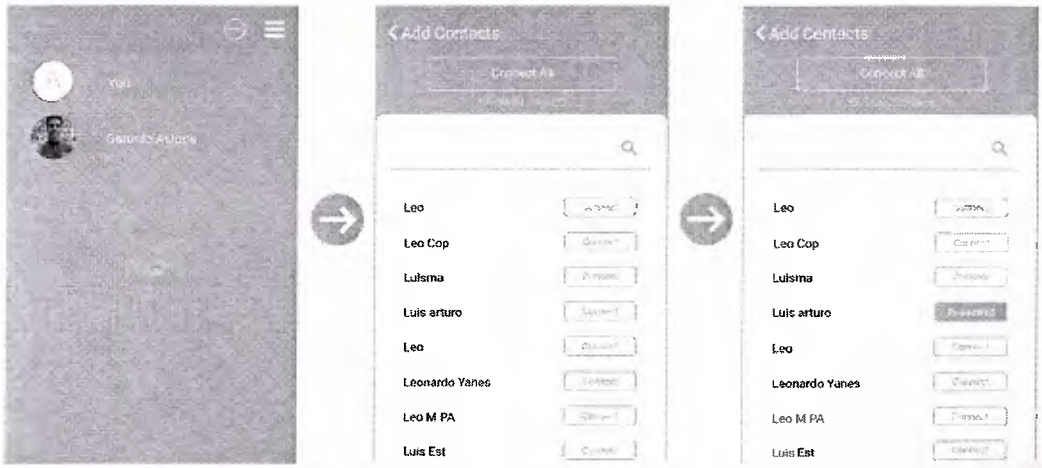

Figura 16. – Proceso de solicitud de conexión con contacto [Elaboración propia]

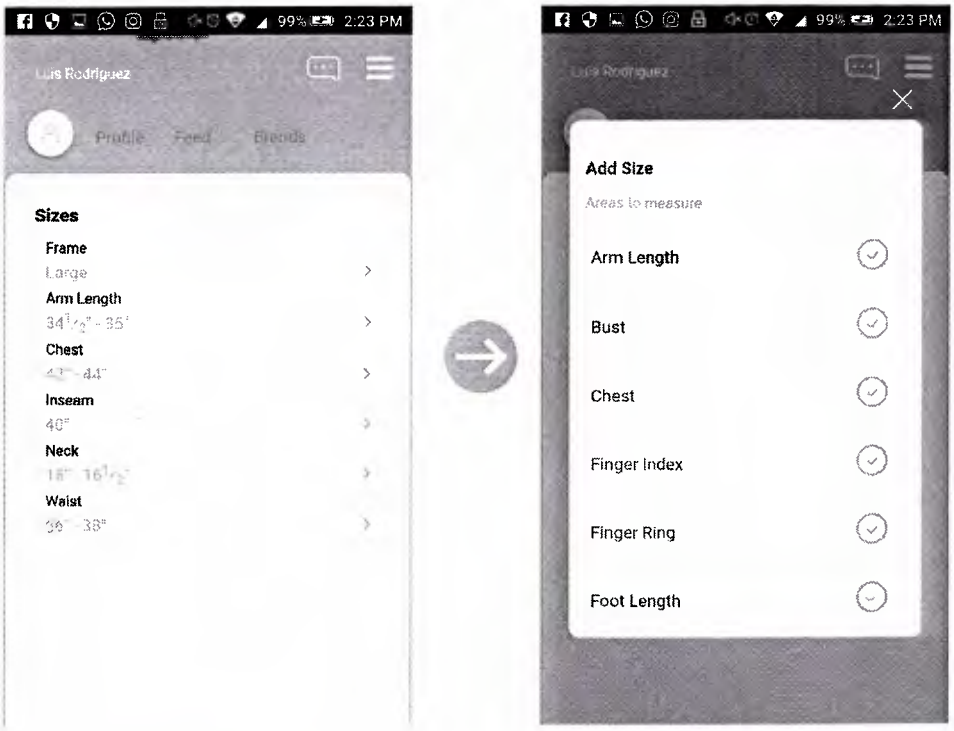

Figura 17. – Solicitud de medida de parte del cuerpo [Elaboración propia]

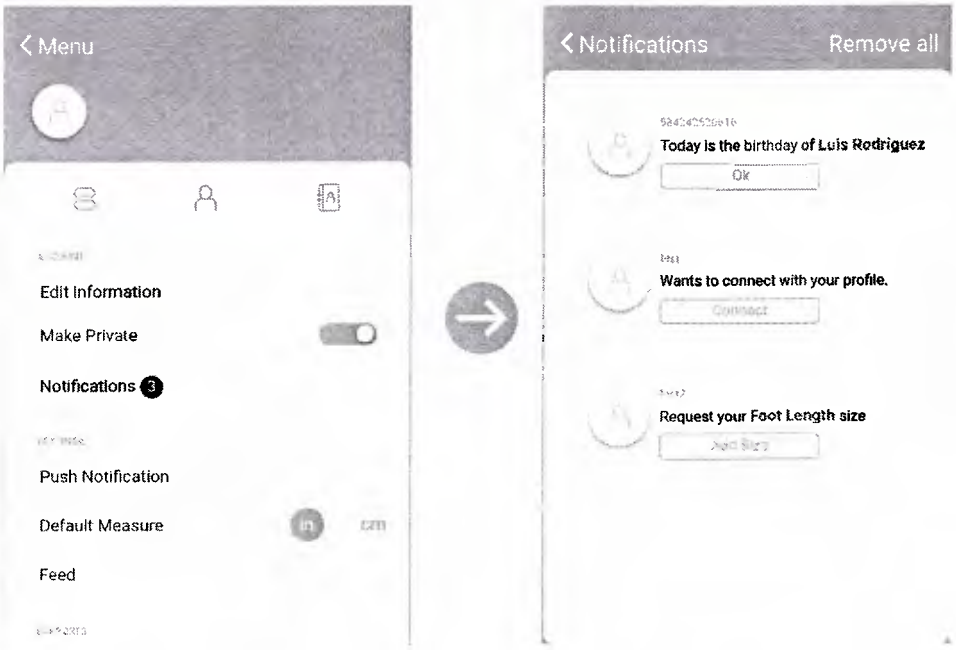

Figura 18. – Visualización de notificaciones [Elaboración propia]

Sprint 5 – Módulo de búsqueda, comparación y emparejamiento

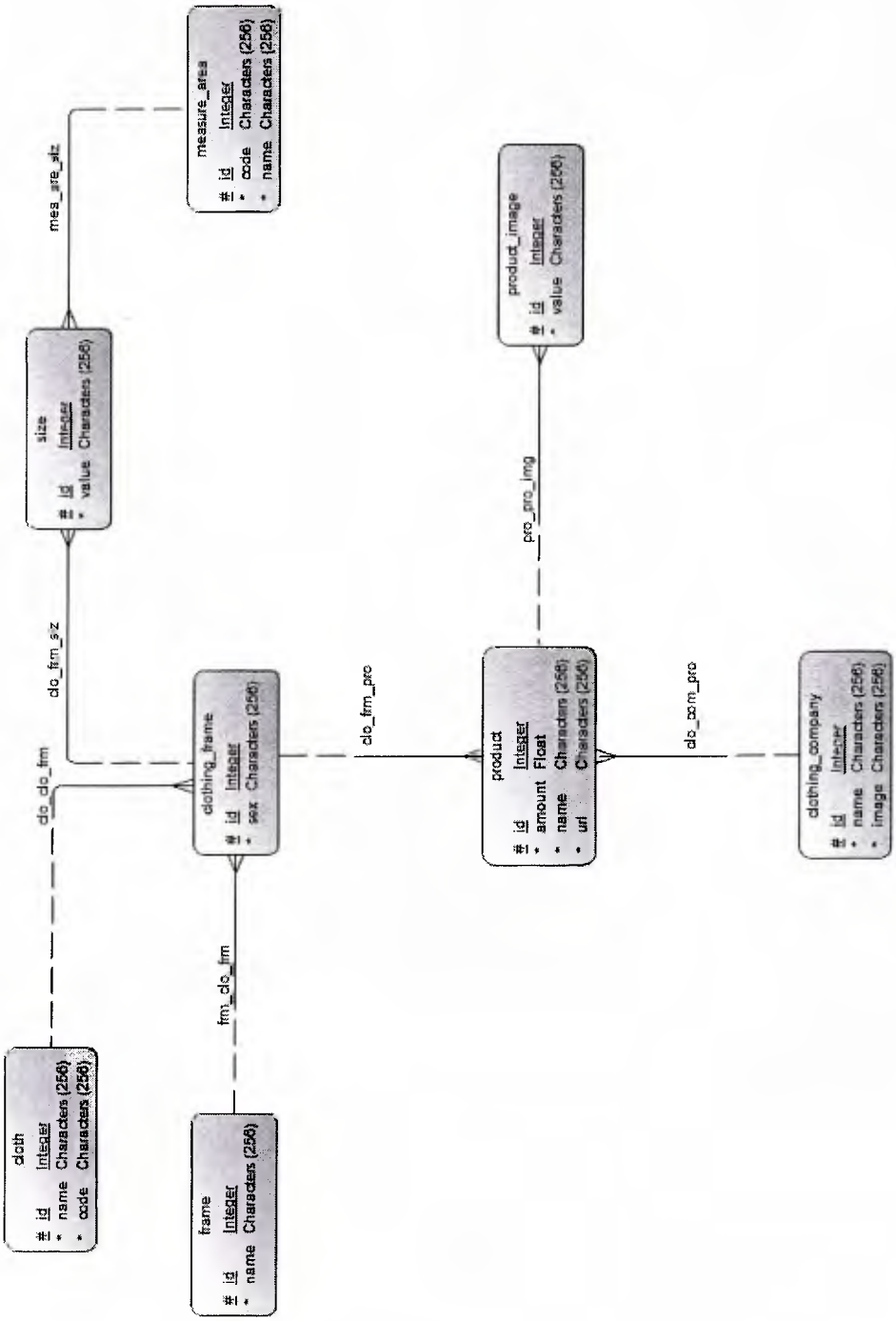

Figura 19 – Modelo de datos MySql [Elaboración propia]

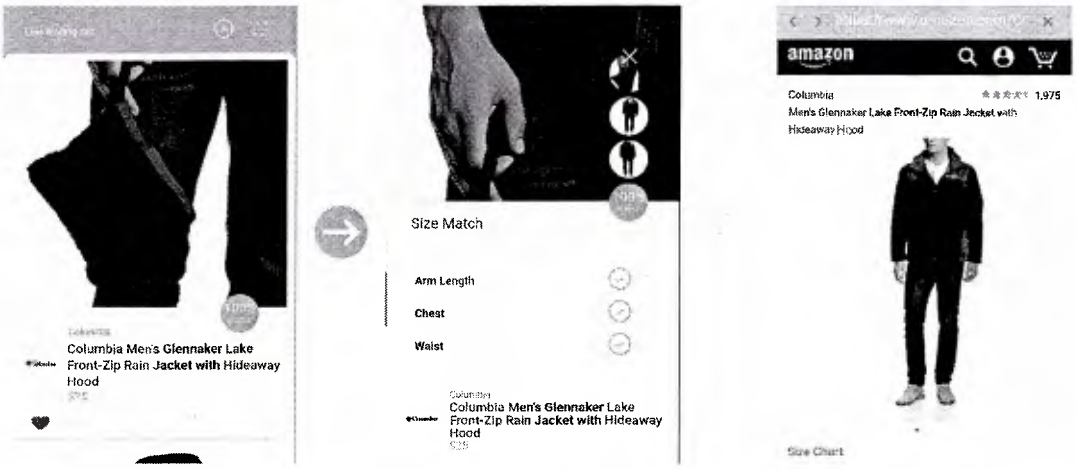

Figura 20 – Consulta de prendas [Elaboración propia]

Sprint 6 – Módulo de Configuraciones y ajustes

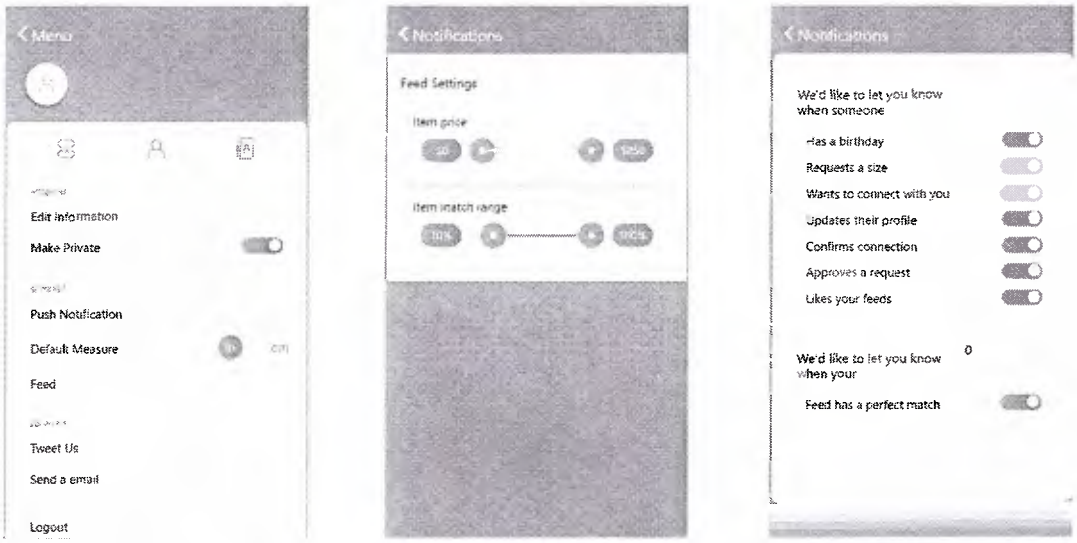

Figura 21. – Configuraciones en el aplicativo móvil [Elaboración propia]