

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

**DESARROLLO DE UN SISTEMA *WEB* Y *MÓVIL* PARA EL
MONITOREO DEL ESFUERZO FÍSICO PERSONALIZADO DE
LOS DEPORTISTAS EN EL CENTRO INTEGRAL DE
ACONDICIONAMIENTO FÍSICO *LIVE SPORT CENTER*.**

**TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN
ANÁLISIS DE SISTEMAS INFORMÁTICOS**

RAUL GEOVANY BASTIDAS GONZÁLEZ

raul.bastidas@epn.edu.ec

TATIANA MAYBETH MONTENEGRO COQUE

tatiana.montenegro@epn.edu.ec

DIRECTOR: Ing. Juan Pablo Zaldumbide, MSc.

juan.zaldumbide@epn.edu.ec

CODIRECTORA: Ing. Mónica Vinueza Rhor, MSc.

monica.vinueza@epn.edu.ec

Quito, Noviembre 2019

DECLARACIÓN

Nosotros, RAÚL GEOVANY BASTIDAS GONZÁLEZ Y TATIANA MAYBETH MONTENEGRO COQUE, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría, que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación –COESC-, somos titulares de la obra en mención y otorgamos una licencia gratuita intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional. Entregaremos toda la información técnica pertinente. En el caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la formativa nacional vigente.

Raúl Geovany Bastidas González

**Tatiana Maybeth Montenegro
Coque**

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por el señor Raúl Geovany Bastidas González y la señorita Tatiana Maybeth Montenegro Coque, bajo nuestra supervisión.

Ing. Juan Pablo Zaldumbide, MSc

DIRECTOR DE PROYECTO

Ing. Mónica Vinuesa Rhor, MSc

CODIRECTORA DE PROYECTO

DEDICATORIA

Quiero dedicar este proyecto principalmente a mi madre Mariana Coque quién siempre estuvo apoyándome incondicionalmente en mis decisiones tanto académicas como deportivas.

TATIANA MAYBETH MONTENEGRO COQUE

AGRADECIMIENTO

Agradezco enormemente al pilar más importante de mi vida, al ser más bondadoso y sensible que he podido conocer, mi mami, Mariana Coque, gracias por tus enseñanzas, sacrificio y cariño incondicional, ¡lo logré mami!, quiero que sepas que el dolor no es para siempre y que todo lo que está pasando nos hará más fuertes, eres muy valiente y estoy muy segura que muy pronto te veré sonriendo y feliz.

A mis hermanos, Junior, Anthony y Jahir por las bromas, los días buenos y los días malos, gracias por hacer de mí cada día una mejor persona.

A mis primos, Emily y Steven que con su voz de aliento nunca dejaron que decaiga en todo este tiempo.

Al Ing. Juan Pablo Zaldumbide por ser la mejor guía para la realización de este proyecto, gracias por los conocimientos, la paciencia y el tiempo dedicado.

TATIANA MAYBETH MONTENEGRO COQUE

RESUMEN

Este proyecto corresponde al desarrollo de un sistema web y móvil para el monitoreo del esfuerzo físico personalizado de los deportistas en el centro integral de acondicionamiento físico *Live Sport Center*. La aplicación móvil será implementada en la plataforma de *Android*.

Se utilizó *Scrum* para el desarrollo de este sistema debido a que con esta metodología se logró llevar un mejor seguimiento de los avances realizados. Este documento se encuentra dividido en cuatro secciones, los cuáles se detallan a continuación.

En la sección I se detalla las actividades que realiza *LSC* y a su vez las falencias que tiene, se especifica el objetivo general, los objetivos específicos y el alcance que tendrá el proyecto.

En la sección II se detalla los roles y conceptos básicos de la metodología *Scrum*, el diseño del sistema y las herramientas que se utilizaron para la implementación del sistema *web* y la aplicación móvil.

En la sección III se detalla cómo se llevó a cabo el desarrollo del sistema *web* y aplicación móvil, además de los resultados obtenidos al realizar las pruebas de compatibilidad y unitarias de cada *Sprint* definido.

Finalmente, la sección IV presenta las conclusiones y recomendaciones obtenidas en el desarrollo de todo el proyecto.

Palabras claves: *Scrum, LSC, Android*

ABSTRACT

This project corresponds to the development of a web and mobile system for monitoring the personalized physical effort of athletes in the Live Fitness Center comprehensive fitness center. The mobile application will be implemented on the Android platform.

Scrum was used for the development of this system because with this methodology it was possible to keep a better track of the progress made. This document is divided into four sections, which are detailed below.

In section I details the activities carried out by LSC and in turn the shortcomings that it has, it specifies the general objective, the specific objectives and the scope that the project will have.

Section II details the basic roles and concepts of the Scrum methodology, the system design and the tools that were used for the implementation of the web system and the mobile application.

Section III details how the development of the web system and mobile application was carried out, in addition to the results obtained when performing the compatibility and unit tests of each defined Sprint.

Finally, section IV presents the conclusions and recommendations obtained in the development of the entire project.

Keywords: Scrum, LSC, Android

ÍNDICE DE CONTENIDOS

DECLARACIÓN	I
CERTIFICACIÓN	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
RESUMEN	V
ABSTRACT	VI
1. INTRODUCCIÓN.....	1
1.1. Objetivo General	1
1.2. Objetivos Específicos	1
1.3. Alcance	2
2. METODOLOGÍA.....	3
2.1. Levantamiento de requerimientos	4
2.2. Análisis de Requerimientos	5
2.2.1. Requerimientos Funcionales	5
2.3. Modelado de la Base de Datos.....	11
2.4. Prototipos del sistema web y aplicación móvil.....	14
2.4.1. Prototipo del Sistema Web	14
2.4.2. Prototipo de la Aplicación Móvil	17
2.5. Diseño de la Arquitectura del Sistema.....	19
2.6. Herramientas para la implementación	19
2.6.1. Herramientas para el acceso a la <i>Firebase Realtime Database</i>	19
2.6.2. Herramientas de Desarrollo <i>Web</i>	20
2.6.3. Herramientas de Desarrollo Móvil	21
3. RESULTADOS Y DISCUSIÓN.....	23
3.1. Desarrollo Web	26
3.2. Desarrollo Móvil	30
3.3. Pruebas de Funcionamiento	33
4. CONCLUSIONES Y RECOMENDACIONES	41
5. REFERENCIAS BIBLIOGRÁFICAS	43
ANEXOS	45

ÍNDICE DE FIGURAS

Figura 2.1 Mapa Mental del MVP del Presente Proyecto	5
Figura 2.2 Casos de Uso Súper Administrador.....	6
Figura 2.3 Casos de Uso Entrenador.....	7
Figura 2.4 Casos de Uso Deportista Parte de LSC.....	8
Figura 2.5 Casos de Uso Deportista Externo.....	9
Figura 2.6 Modelo Entidad Relación del Presente Sistema	11
Figura 2.7 Página Principal del Sistema Web	15
Figura 2.8 Login del Sistema Web	15
Figura 2.9 Menús Pantalla Super Administrador.....	16
Figura 2.10 Menús Pantalla Super Administrador.....	16
Figura 2.11 Slides de Introducción.....	17
Figura 2.12 Formulario de Registro de un Usuario Deportista Externo	17
Figura 2.13 Pantalla de Inicio de Sesión.....	18
Figura 2.14 Menús y Pantalla Home	18
Figura 2.15 Diagrama de Infraestructura	19
Figura 3.1 Esquema de la base de datos del sistema	25
Figura 3.2 Tabla en Firebase de usuarios autenticados	26
Figura 3.3 Sistema Web visto desde el navegador Firefox.....	35
Figura 3.4 Visualización de la app móvil en ionic lab.....	36
Figura 3.5 Visualización en plataforma Android ionic lab.....	36
Figura 3.6 Vista del proyecto FitApp en DevApp	37
Figura 3.7 Pantalla de la aplicación FitApp en Ionic DevApp.....	37
Figura 3.8 Prueba de interfaz: elementos gráficos.....	38
Figura 3.9 Prueba de interfaz: tipografías y textos.....	39
Figura 3.10 Datos de entrada y acciones al registrarlo.....	39
Figura 3.11 Datos de entrada para iniciar sesión y acciones al iniciar	40
Figura 3.12 Mensajes de alerta en la aplicación móvil.....	40
Figura 3.13 Mensaje al desear ingresar a la galería de fotos del dispositivo....	41

ÍNDICE DE TABLAS

Tabla 2.1 Equipo de trabajo y asignación de roles	4
Tabla 2.2 Requerimientos Funcionales del sistema web	5
Tabla 2.3 Requerimientos Funcionales de la aplicación móvil.....	6
Tabla 3.1 Resumen de historias de usuario	23
Tabla 3.2 Resumen de Sprints	24
Tabla 3.3 Pruebas de Caja Negra en el Sistema Web.....	33
Tabla 3.4 Navegadores que soporta el Sistema FitApp.....	35
Tabla 3.5 Pruebas de compatibilidad en Smartphones.....	38

1. INTRODUCCIÓN

Planteamiento del problema

Live Sport Center (LSC) es un centro de acondicionamiento físico que tiene como misión brindar un servicio de actividad física esmerando la atención, y ofreciendo gran variedad de actividades y servicios, a sus clientes, pero en la actualidad, no cuenta con tecnología que permita registrarlos y dar seguimiento a la actividad realizada por cada cliente.

LSC al no disponer de medios y procesos para ayudar al cliente a conocer oportunamente el logro de sus objetivos (pérdida de peso y quemar grasa, ganar masa muscular y fuerza, vivir de una forma saludable y mantener peso), hace que los servicios que presta sean poco ágiles, tal como se evidencia en el tiempo que tardan los entrenadores para entregar a sus clientes los resultados de su esfuerzo físico, que, al hacerlo manualmente, este lleva entre 3 o 4 semanas. Además, datos estadísticos demuestran que el aumento de la cultura por el deporte es cada día más alto lo cual agrava esta situación (El Telégrafo, 2015).

Por lo anteriormente expuesto, se propone desarrollar un sistema *web* y móvil que permita llevar el control permanente de los deportistas y brindarles información de ejercicios y recetas además de monitorear la evolución del entrenamiento del deportista y consultar mediante gráficos estadísticos el progreso que ha tenido.

1.1. Objetivo General

Desarrollar un sistema *web* y móvil para el monitoreo del esfuerzo físico personalizado de los deportistas en el centro integral de acondicionamiento físico *Live Sport Center*.

1.2. Objetivos Específicos

- Determinar los requerimientos del sistema
- Diseñar el modelo de la base de datos
- Diseñar prototipos del sistema *web* y móvil
- Desarrollar el sistema *web*.
- Desarrollar la aplicación móvil.
- Realizar pruebas de funcionamiento y usabilidad.

1.3. Alcance

Los centros de acondicionamiento físico y de cuidado personal deben adaptarse y ser cada vez más novedosos en la prestación de sus servicios para lograr de manera eficaz cumplir con los objetivos de los clientes.

Por lo tanto, el presente trabajo permitió automatizar y controlar las actividades relacionadas con la práctica de un deporte, una necesidad imperiosa para LSC.

A la vez el sistema *web* permitió al super administrador actuar como un gestor de contenido para la página principal, es decir, le permitió añadir, modificar y eliminar contenido en las secciones de inicio, acerca de, servicios, portafolio y contacto lo que mantuvo al usuario siempre informado con lo más actual de LSC, además facultó al administrador gestionar recetas, entrenamientos, rutinas, su perfil y deportistas en donde fue capaz de realizar el seguimiento de las actividades que cada uno realizó mediante gráficos estadísticos.

Mientras que la aplicación móvil permitió al usuario deportista gestionar su perfil, acceder a recetas además de poder seguir diferentes entrenamientos compuestas por diferentes rutinas disponibles. También posibilitó el cálculo del IMC, peso ideal, calorías diarias, calorías quemadas y finalmente con ayuda de gráficos estadísticos ver el progreso de cada usuario.

2. METODOLOGÍA

Metodología de Desarrollo **SCRUM**

Scrum es un entorno de trabajo creado por *Ken Schwaber* y *Jeff Sutherland* para desarrollar, entregar y mantener resultados complejos, en el que se aplican diversos métodos y procesos, de manera que el producto, el equipo y el entorno de trabajo reciban mejoras continuas (*Schwaber & Sutherland, 2017*).

Roles

Personas que intervienen de manera directa o indirecta con el proyecto (Palacio, 2015).

Scrum propone los siguientes roles:

Product Owner

Persona responsable de priorizar y gestionar las tareas del *Product Backlog*, además de ser intermediario entre el cliente y equipo de desarrollo (Requena Mesa, 2018).

Scrum Master

De acuerdo a Requena Mesa (2018), es el lazo entre *Product Owner* y el *Development Team*, encargado de mantener motivada a la gente de trabajo además de revisar y validar el *Product Backlog*.

Development Team

Grupo o grupos de trabajo que desarrollan el producto. Según Palacio (2015), en el *Development Team*:

- Todos conocen y entienden la visión del *Product Owner*.
- Contribuyen con el *Product Owner* en el desarrollo del *Product Backlog*.
- Participan en las decisiones.

Asignación de Roles

La asignación de roles en el equipo de trabajo del sistema *web* y móvil se estableció de la siguiente manera. Ver Tabla 2.1.

Tabla 2.1 Equipo de trabajo y asignación de roles

Miembros	Roles
Lic. Alberto Orozco	<i>Product Owner</i>
Ing. Juan Pablo Zaldumbide Proaño. Msc	
Ing. Mónica Vinueza Rhor. Msc	<i>Scrum Master</i>
Raúl Bastidas, Tatiana Montenegro	<i>Development Team</i>

Artefactos

Los artefactos de *scrum* simbolizan trabajo que debe estar definido de manera clara de modo que todos los miembros *scrum* comprendan y sepan a donde se dirige el proyecto.

Product Backlog

Lista de requisitos o historias de usuario del cliente, organizadas de acuerdo a la prioridad que les da el mismo, a la vez esta lista crece y progresa durante el desarrollo (Palacio, 2015) .

Sprint Backlog

Es la lista formada por tareas en las que se divide el *Product Backlog* (historias de usuario) que se van a poner en marcha para construir una parte completa y funcional del producto (Palacio, 2015)

User Stories

Elemento del *Product Backlog* que proporcionan información de cómo debe ser el comportamiento del requerimiento que se trabaja, en caso de cambio brinda información directa del cliente (Lara, 2015)

2.1. Levantamiento de requerimientos

El levantamiento de requerimientos se realizó de acuerdo a la compilación de información que se obtuvo de las reuniones con el entrenador personal de LSC y los desarrolladores.

Para ello se decidió utilizar un mapa mental¹ en donde se plasmó ideas y necesidades del entrenador en base al presente proyecto, esto permitió crear un MVP² del mismo. Ver Figura 2.1.

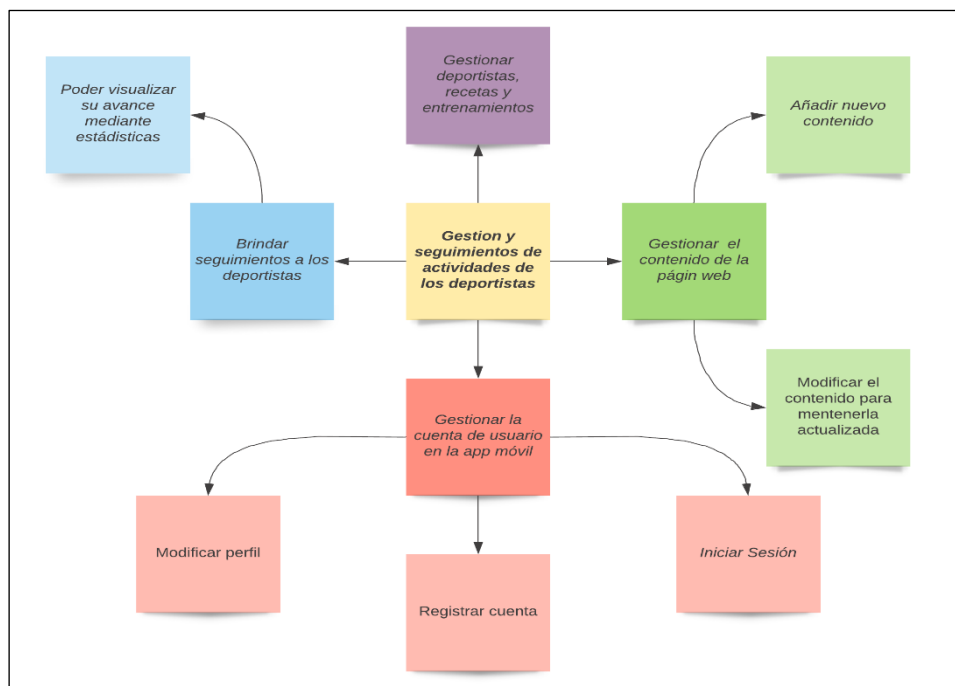


Figura 2.1 Mapa Mental del MVP del Presente Proyecto

2.2. Análisis de Requerimientos

2.2.1. Requerimientos Funcionales

Los requerimientos funcionales describen las actividades y servicios que un sistema debe proveer. Normalmente este tipo de requerimientos están asociados con las entradas, las salidas de los procesos y los datos a almacenar en el sistema, como se muestra en la Tabla 2.2 y Tabla 2.3 (Villarroel, 2017).

Tabla 2.2 Requerimientos Funcionales del sistema web

Id	Funcionalidad
RFW1	Inicio y cierre de sesión del usuario super administrador
RFW2	Modificación del <i>One Page</i>
RFW3	Modificación del usuario administrador
RFW4	Inicio y cierre de sesión del usuario entrenador

¹ **Mapa mental:** diagrama que permite representar distintos conceptos, ideas, tareas, etc dispuestos gráficamente de distintas maneras alrededor de una palabra clave o concepto central. Obtenido de: <https://concepto.de/mapa-mental/>.

² **MVP:** (Producto Mínimo Viable) es una versión parcial de un producto orientada a descubrir rápidamente qué pide el cliente. Obtenido de: <https://www.inboundcycle.com/blog-de-inbound-marketing/bid/189238/Qu-es-el-MVP-o-Producto-M-nimo-Viable>.

Id	Funcionalidad
RFW5	Modificar perfil administrador
RFW6	Registro, modificación y eliminación de usuario deportista
RFW7	Registro, modificación y eliminación de recetas
RFW8	Registro, modificación y eliminación de ejercicios
RFW9	Registro, modificación y eliminación de rutinas
RFW10	Registro, modificación y eliminación de entrenamientos
RFW11	Visualizar avance del usuario deportista mediante gráficos estadísticos

Tabla 2.3 Requerimientos Funcionales de la aplicación móvil

Id	Funcionalidad
RFM1	Registro del usuario
RFM2	Inicio y cierre de sesión mediante credenciales, <i>Facebook</i> o <i>Gmail</i>
RFM3	Modificar perfil usuario
RFM4	Visualizar información de recetas, ejercicios, rutinas del api generada de la página <i>web</i> .
RFM5	Calculadora de imc, peso ideal, calorías diarias y quemadas
RFM6	Visualizar avance del usuario mediante estadísticas

Casos de Uso

De acuerdo a los requerimientos funcionales y al *Product Backlog*, se crearon diagramas de casos de uso³ tanto del Sistema *Web* como de la Aplicación Móvil, tal como se muestra en las figuras a continuación.

Diagramas Casos de Uso Sistema *Web*

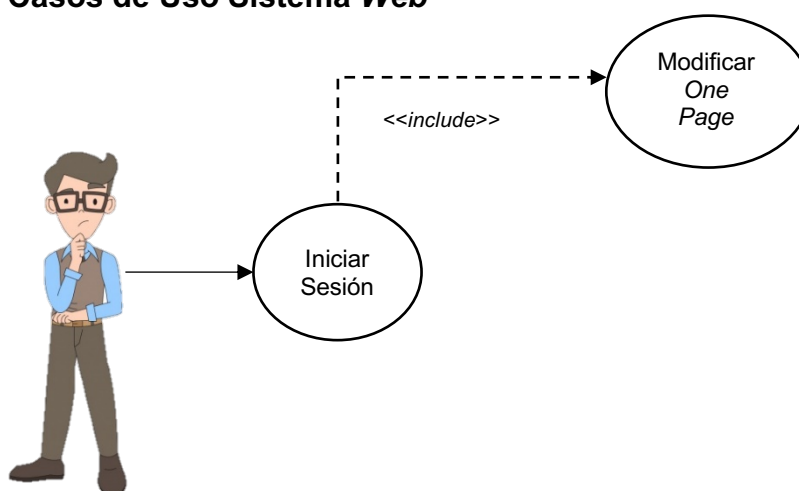


Figura 2.2 Casos de Uso Súper Administrador

³ **Diagrama casos de uso:** diagramas muestran operaciones que se esperan de sistema y como se relaciona con su entorno, es por ello que se ve desde el punto de vista del usuario. Obtenido de <http://ingenieriadesistemas-shirley.blogspot.com/2012/05/tipos-de-diagramas-uml.html>

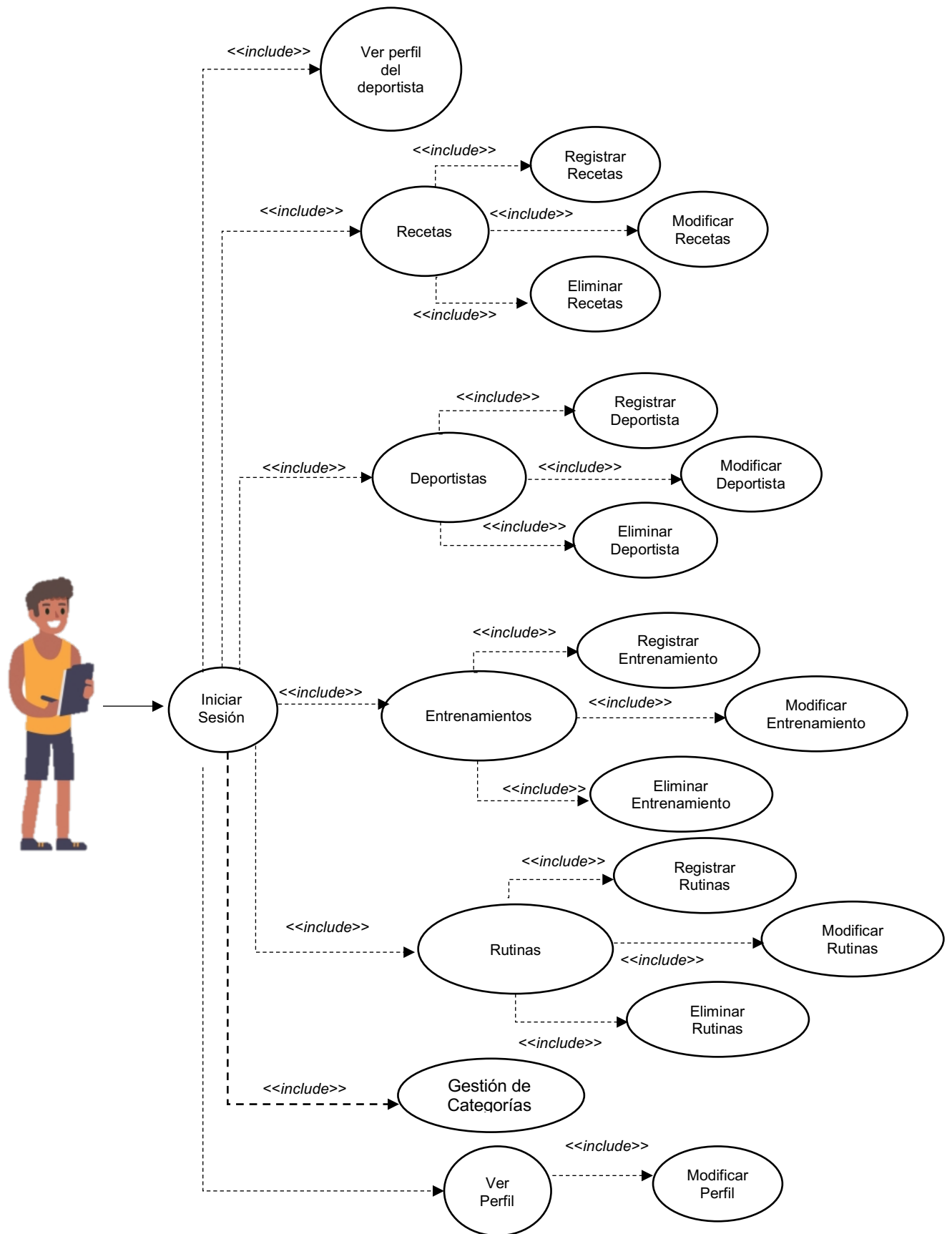


Figura 2.3 Casos de Uso Entrenador

Diagramas Casos de Uso Aplicación Móvil

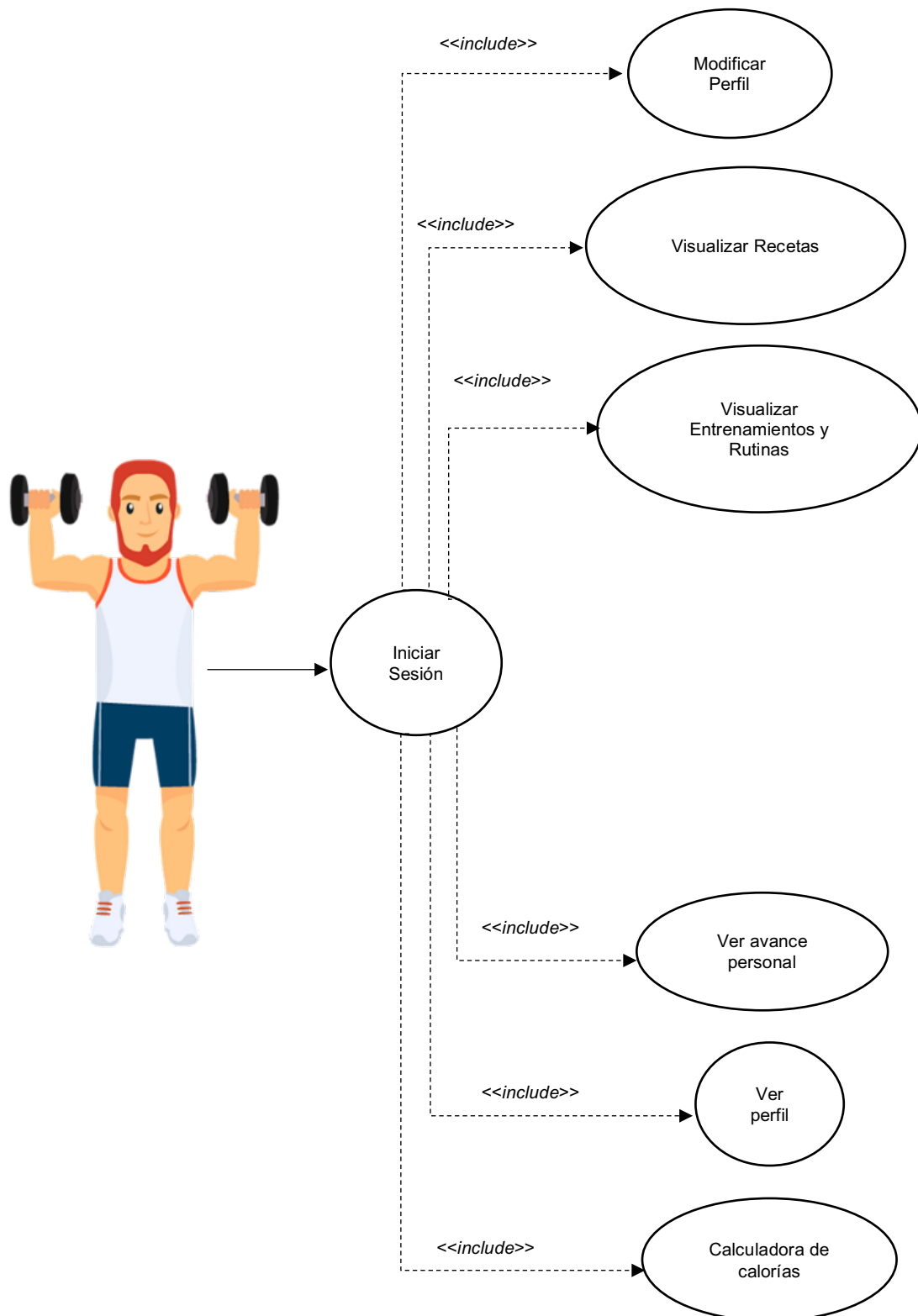


Figura 2.4 Casos de Uso Deportista Parte de LSC

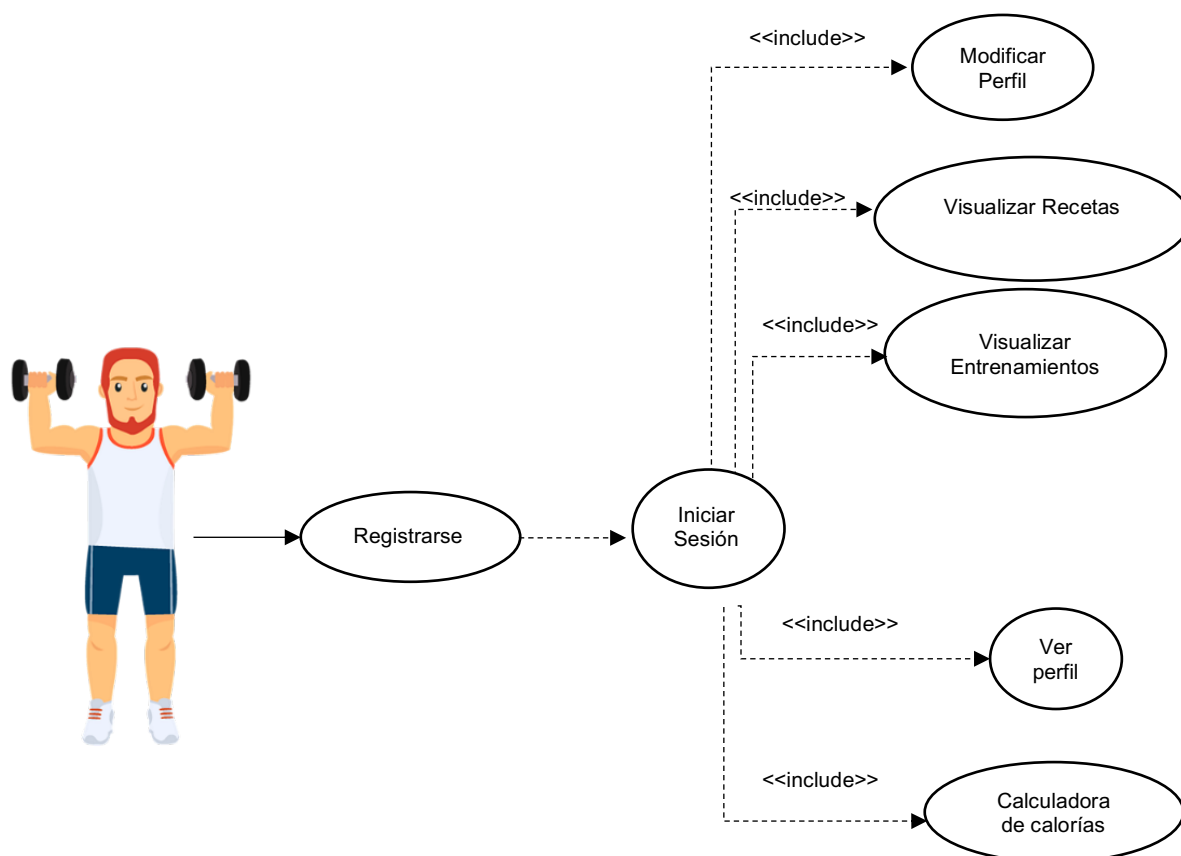


Figura 2.5 Casos de Uso Deportista Externo

De acuerdo a Cevallos (2015), un diagrama de casos de uso se forma de los siguientes símbolos y relaciones.

- **Casos de Uso:** se representan con óvalos, la etiqueta en el óvalo indica la función del sistema.
- **Actor:** Se refiere a un rol específico de un usuario del sistema.
- **Relaciones:** las relaciones entre un actor y casos de uso se representan con una línea simple. Para relaciones entre casos de uso se utilizan flechas etiquetadas “*include*” o “*extend*”. Una relación “*include*” indica que un caso de uso es necesitado por otro para poder cumplir una tarea. Una relación “*extend*” indica opciones alternativas para un cierto caso de uso.

Usuarios del Sistema

Se identificaron cuatro tipos de usuarios, los cuales se detallan a continuación.

- **Súper Administrador:** hace referencia a los desarrolladores del sistema, es el usuario que tiene los permisos para añadir, modificar o eliminar contenido en el *one page*. Además, puede modificar la información del usuario entrenador disponible en la sección contacto. Este usuario debe autenticarse en el sistema web mediante sus credenciales.
- **Entrenador:** hace referencia al administrador del sistema *web* capaz de registrar, modificar, o eliminar deportistas, recetas, ejercicios, rutinas y entrenamientos. Adicionalmente, es capaz de actualizar su perfil de usuario y dar un seguimiento al avance de sus deportistas accediendo a sus perfiles. Para acceder al sistema *web* lo hace mediante credenciales proporcionadas por el Súper Administrador.
- **Deportista:** el usuario deportista hace uso de la aplicación móvil, se han identificado dos tipos de deportistas.

Deportista parte de LSC: es el usuario registrado por el entrenador en el sistema *web*, recibe las credenciales por parte del entrenador para poder autenticarse en la aplicación móvil. En la aplicación móvil puede modificar su perfil, visualiza y accede a recetas, entrenamientos compuestos por rutinas de ejercicios, además de poder ver sus estadísticas personales. Puede también calcular su IMC, peso ideal, calorías diarias y calorías quemadas accediendo al apartado calculadora.

Deportista externo: es el usuario que no es registrado por el entrenador en el sistema. Este tipo de usuario debe registrarse por medio del formulario, *Facebook* o *Gmail* para poder iniciar sesión en la aplicación móvil. Puede calcular su IMC, peso ideal, calorías diarias y calorías quemadas accediendo al apartado calculadora, modificar su perfil además de acceder y visualizar recetas y ejercicios generales.

2.3. Modelado de la Base de Datos

En el presente proyecto se utilizaron bases de datos no relacionales, pero para obtener mayor abstracción se decidió modelar un esquema entidad relación que permitió mostrar datos relevantes del sistema, así como relaciones entre ellos, como se muestra en la Figura 2.6.

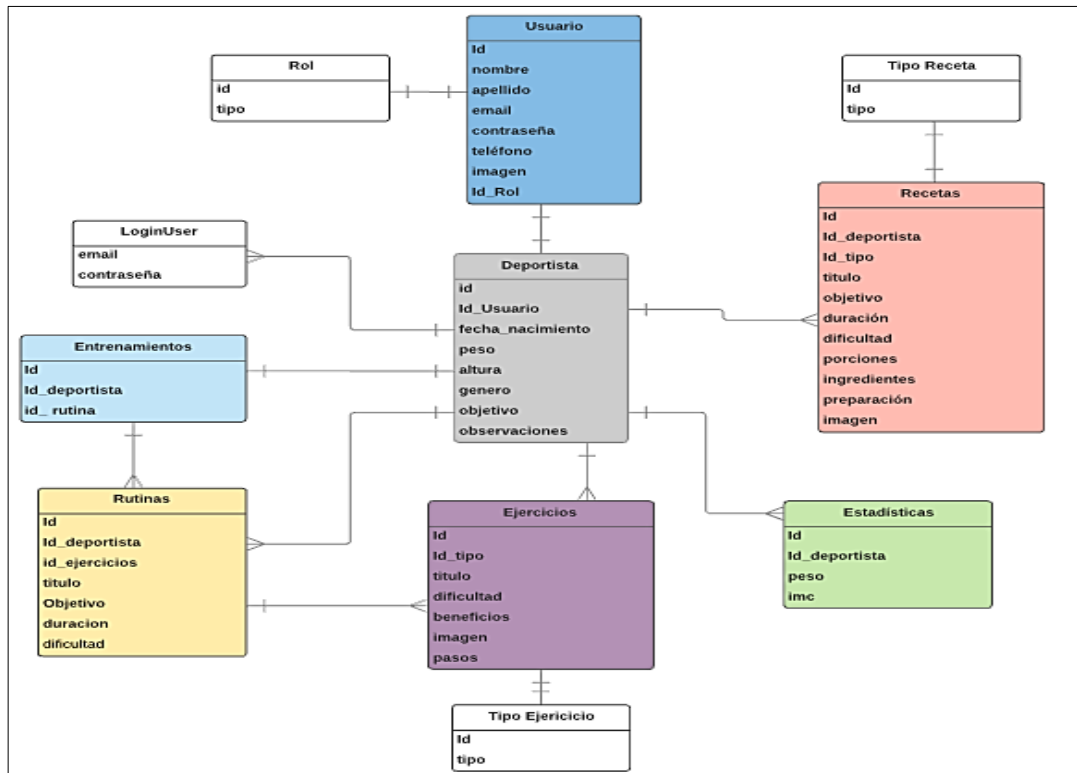


Figura 2.6 Modelo Entidad Relación del Presente Sistema

Tabla Usuarios: almacena los datos de los usuarios que van a utilizar el sistema web y la aplicación móvil. Los atributos que posee son:

- **Id:** (identificador único del usuario) permite controlar a los usuarios registrados.
- **Nombre:** almacena el nombre del usuario
- **Apellido:** almacena el apellido del usuario
- **Email:** almacena el correo electrónico que permitirá el ingreso al sistema web o aplicación móvil.
- **Contraseña:** en caso de ser deportista parte de LSC se genera a partir del correo electrónico del mismo, mientras que si es deportista externo almacena la contraseña registrada por el mismo.
- **Teléfono:** almacena el teléfono del usuario deportista parte de LSC.

- **Imagen:** almacena una imagen por defecto, modificable en la aplicación móvil.
- **Id Rol:** almacena el rol del usuario, permite identificar si se trata de un super administrador, administrador para el sistema *web*, y deportista parte de *LSC* o deportista externo en la aplicación móvil.

Tabla Deportistas: almacena los datos de los deportistas que utilizarán la aplicación móvil. Los atributos que posee son:

- **Id:** (identificador único del usuario deportista) permite controlar a los deportistas registrados.
- **Id Usuario:** identificador único del usuario, permite hacer referencia a los datos que el usuario con este id almacena.
- **Fecha nacimiento:** almacena la fecha de nacimiento del deportista.
- **Peso:** almacena el peso del deportista comprendido entre 30 y 200 kg
- **Altura:** almacena la altura del deportista comprendido entre 130 y 220 cm.
- **Género:** almacena el género (hombre o mujer) del deportista.
- **Objetivo:** almacena el objetivo del deportista (perder peso y quemar grasa, ganar masa muscular y fuerza, vivir de forma saludable y mantener mi peso).
- **Observaciones:** almacena datos relacionados a problemas de salud del deportista.

Tabla Recetas: almacena las recetas que serán filtradas en la aplicación móvil de acuerdo al objetivo del usuario deportista. Los atributos que posee son:

- **Id:** (identificador único de la receta) permite controlar las recetas registradas.
- **Id deportista:** permite acceder al objetivo del deportista para poder filtrar las recetas.
- **Id tipo:** almacena el tipo de receta.
- **Título:** almacena el nombre de la receta.
- **Objetivo:** almacena la descripción, objetivo o detalle de la receta
- **Duración:** almacena el tiempo que toma preparar la receta.
- **Dificultad:** almacena la dificultad (media, baja, alta).

- **Porciones:** almacena las porciones que rinde la receta.
- **Ingredientes:** almacena los ingredientes necesarios para preparar la receta.
- **Preparación:** almacena los pasos a seguir para preparar la receta.
- **Imagen:** almacena una foto de la receta ya preparada.

Tabla Ejercicios: almacena los ejercicios que estará disponibles en la aplicación móvil y que componen las rutinas. Los atributos que posee son:

- **Id:** (identificador único de los ejercicios) permite controlar los ejercicios registrados.
- **Id tipo:** almacena el tipo de ejercicio.
- **Título:** almacena el nombre del ejercicio.
- **Dificultad:** almacena la dificultad (principiante, medio o avanzado).
- **Beneficios:** almacena descripción de los beneficios del ejercicio.
- **Pasos:** almacena los pasos a seguir para realizar el ejercicio.
- **Imagen:** almacena una imagen del ejercicio.

Tabla Rutinas: almacena datos sobre rutinas compuestas por ejercicios que serán visibles en la aplicación móvil. Los atributos que posee son:

- **Id:** (identificador único de la notificación) permite controlar las promociones registradas.
- **Id deportista:** permite identificar al deportista parte de *LSC* para poder asignarle ejercicios a su rutina personal.
- **Título:** almacena el nombre de la rutina.
- **Objetivo:** almacena la descripción, objetivo o detalle de la rutina.
- **Duración:** almacena el tiempo de duración de la rutina.

Tabla Entrenamientos: almacena datos sobre las rutinas a realizar por un deportista. Los atributos que posee son:

- **Id:** (identificador único de la rutina) permite controlar las rutinas registradas.
- **Id deportista:** permite acceder a los datos del deportista registrado.
- **Id rutina:** permite acceder a los datos de la rutina registrada.

Tabla Estadísticas: almacena datos sobre los cuales se realizarán los gráficos estadísticos. Los atributos que posee son:

- **Id:** (identificador único de la estadística) permite controlar las estadísticas registradas.
- **Id deportista:** permite acceder a los datos de deportista que servirán de base para realizar las estadísticas.
- **Peso:** almacena el progreso del peso del deportista.
- **IMC:** almacena el progreso del imc del deportista.

Tabla Login User: almacena las credenciales para poder iniciar sesión. Los atributos que posee son:

- **Email:** almacena el correo electrónico necesario para iniciar sesión.
- **Contraseña:** necesaria para iniciar sesión.

Tabla Tipo de Receta: los atributos que posee son:

- **Id:** (identificador único del tipo de receta) permite controlar los tipos de receta registrados.
- **Tipo:** almacena el tipo de receta.

Tabla Tipo de Ejercicio: los atributos que posee son:

- **Id:** (identificador único del tipo de ejercicio) permite controlar los tipos de ejercicio registrados.
- **Tipo:** almacena el tipo de ejercicio.

2.4. Prototipos del sistema web y aplicación móvil

A continuación, se presentan los prototipos de las pantallas principales tanto del sistema *web* como de la aplicación móvil, para mayor detalle ver Anexo C y Anexo D.

2.4.1. Prototipo del Sistema Web

Es la única página principal del sistema *web* en donde el usuario puede ver contenido actualizado acerca del centro *Live Sport Center*, a la vez este contenido fue gestionado por el super administrador. Ver Figura 2.7.

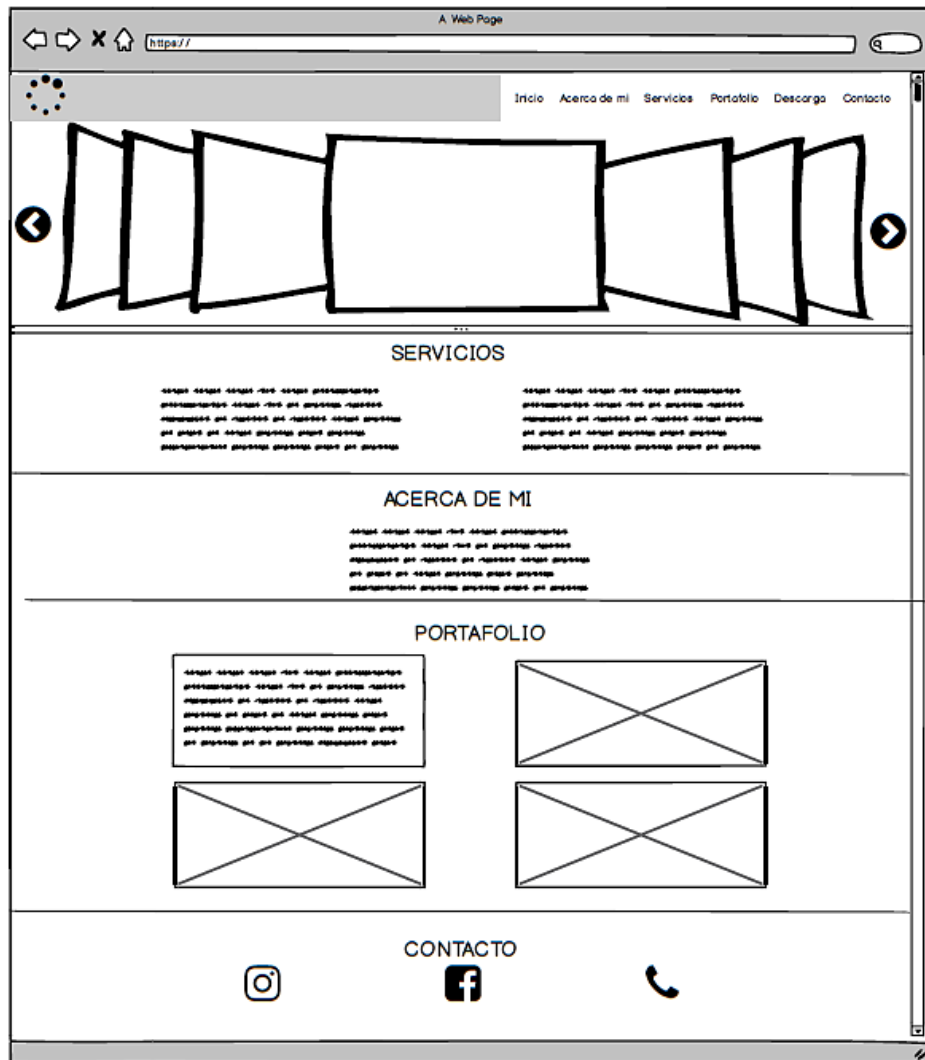


Figura 2.7 Página Principal del Sistema Web

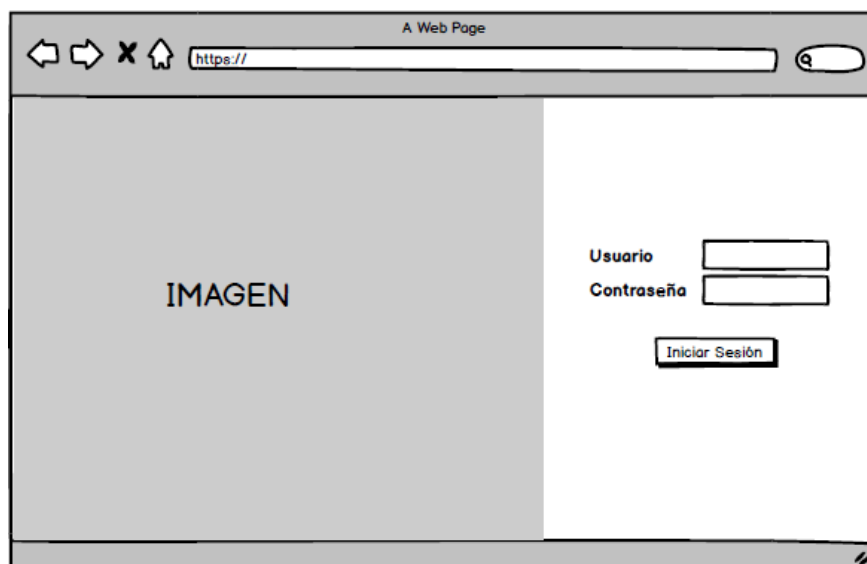


Figura 2.8 Login del Sistema Web

En la Figura 2.8 se muestra la pantalla de inicio de sesión tanto para administrador y super administrador, las credenciales de ambos son proporcionadas por los desarrolladores del sistema.

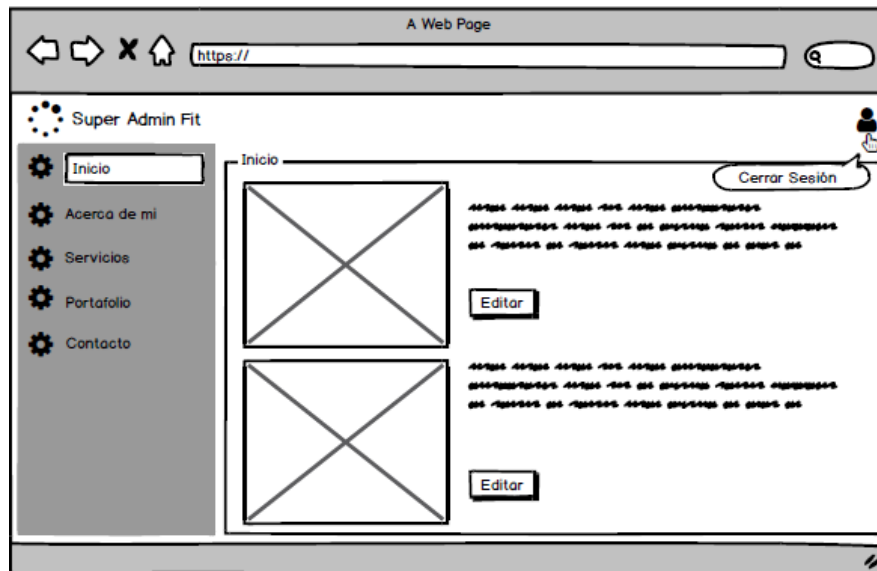


Figura 2.9 Menús Pantalla Super Administrador

La Figura 2.9 se observa la pantalla principal al iniciar sesión en el sistema *web* como super administrador, tiene un menú al lado izquierdo que permitió gestionar el contenido del *one page*.

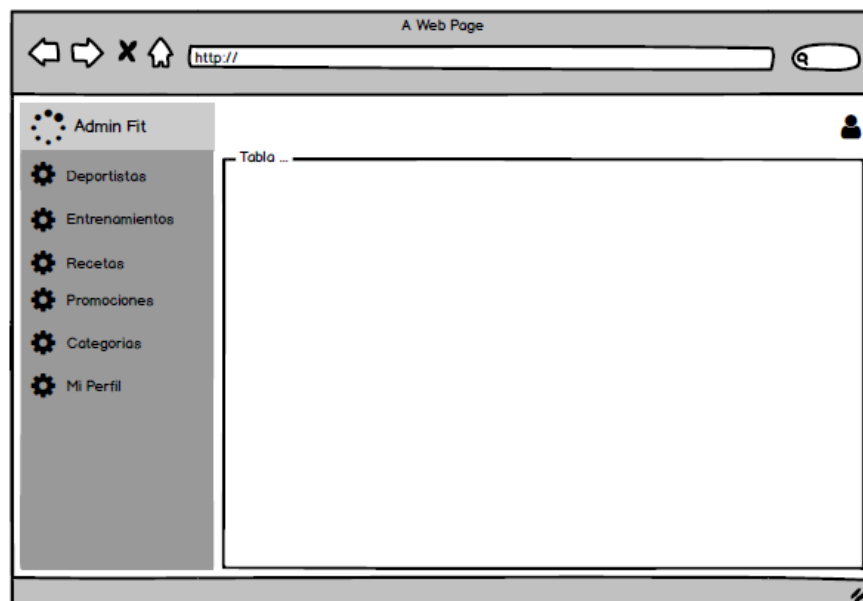


Figura 2.10 Menús Pantalla Super Administrador

La Figura 2.10 muestra la pantalla principal al iniciar sesión en el sistema *web* como administrador, tiene un menú al lado izquierdo que permitió gestionar deportistas, recetas, entrenamientos, rutinas, categorías y perfil administrador.

2.4.2. Prototipo de la Aplicación Móvil



Figura 2.11 Slides de Introducción

Los *sliders* de la Figura 2.11 describen brevemente las características principales de la aplicación móvil.

Four mobile app screens for user registration. The first screen is titled 'GENERO' and shows male and female icons. The second screen is titled 'OBJETIVO' and shows 'SUBIR PESO' with an Apple icon, 'BAJAR PESO' with an Android icon, and 'MANTENER PESO' with a peace sign icon. The third screen is titled 'INFORMACION' and shows a form with fields for 'NOMBRE', 'APELLIDO', 'CORREO', 'CONTRASEÑA', and 'CONFIRMAR CONTRASEÑA'. The fourth screen is titled 'DATOS PERSONALES' and shows a form with fields for 'FECHA DE NACIMIENTO', 'PESO', and 'ALTURA', followed by a 'REGISTRAR' button.

Figura 2.12 Formulario de Registro de un Usuario Deportista Externo

Las pantallas de la Figura 2.12 permitieron el registro de una cuenta para un deportista externo al centro LSC.

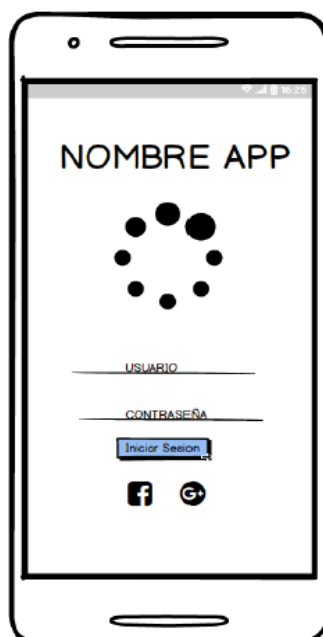


Figura 2.13 Pantalla de Inicio de Sesión

La pantalla de inicio de sesión que muestra la Figura 2.13 cuenta con dos formas de autenticación, es decir, por medio de redes sociales (*facebook* o *gmail*) y también por *email* y contraseña.

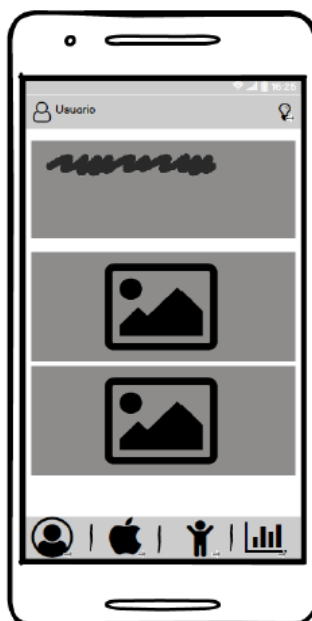


Figura 2.14 Menús y Pantalla *Home*

Posterior al inicio de sesión la primera pantalla que se visualiza es el *Home* como se observa en la Figura 2.14, que contiene los menús principales para acceder a recetas, entrenamientos, estadísticas, además de notificaciones y perfil de usuario.

2.5. Diseño de la Arquitectura del Sistema

La arquitectura tiene como base principal el servidor en la nube Firebase que funciona como nuestra base de datos. El sistema web y la aplicación móvil realizan peticiones *HTTP* al api proveniente de Firebase que nos da acceso a la base de datos alojada para poder guardar u obtener datos en tiempo real de la misma. Ver Figura 2.15.

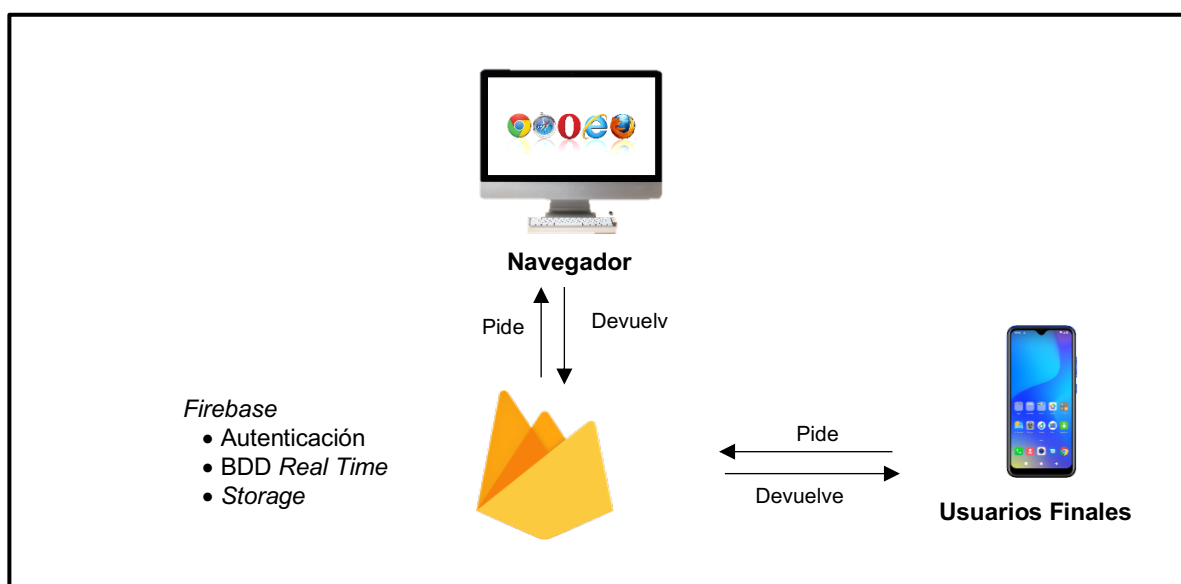


Figura 2.15 Diagrama de Infraestructura

2.6. Herramientas para la implementación

2.6.1. Herramientas para el acceso a la *Firebase Realtime Database*.

Firebase Realtime Database

Firebase Realtime Database es una base de datos alojada en la nube. Los datos se almacenan en formato JSON y se sincronizan en tiempo real con cada cliente conectado (*Firebase*, s.f.).

Esta herramienta posibilitó obtener datos actualizados tanto en el sistema web como en la aplicación móvil gracias a su capacidad de sincronización en tiempo real.

Firestore Authentication

Facilita la creación de sistemas de autenticación seguros que admite cuentas de correo electrónico y contraseña, autenticación de teléfono e inicio de sesión de *Google*, *Twitter*, *Facebook* y *GitHub* (Firestore, s.f.).

Esta herramienta permitió que los usuarios accedan a la aplicación móvil para poder conocer su identidad, facilitando el almacenamiento de datos en la nube de forma segura.

2.6.2. Herramientas de Desarrollo Web

Angular

Angular es un *framework front end* de código abierto para aplicaciones web, escrito en *Typescript* y construido por los desarrolladores de *Google* (Robles, 2017). Permite utilizar *Html* como idioma de su plantilla y le permite extender la sintaxis *Html* para expresar los componentes de su aplicación de manera clara y sucinta (AngularJS, 2018).

Además de su compatibilidad con *Typescript*, otro beneficio que proporcionó al momento de codificar fueron las funcionalidades que ofrece a diferencia de otros softwares en donde por ejemplo hay que acudir a bibliotecas de terceros al momento de ordenar el código.

Angular Material

Es una biblioteca de componentes *UI*⁴ para Angular, que permiten construir páginas y aplicaciones web atractivas, consistentes y funcionales, creando sitios web más rápidos, hermosos y receptivos. Basado en la guía de diseño de *Google Material Design*. (Tutorials Point, s.f.).

⁴ **UI:** *User Interface*, se refiere a la creación de la interfaz. Obtenido de blog.eltallerweb.com/diferencias-entre-ui-y-ux/.

Esta herramienta facilitó la creación de las pantallas que conforman el sistema *web* haciendo que éstas sean amigables e intuitivas al momento de su uso.

Typescript

Es un lenguaje de programación de código abierto desarrollado por Microsoft llamado también *superset*⁵ de *Javascript* lo que significa que, si el navegador está basado en *Javascript*, este nunca llegará a saber que el código original fue realizado con *Typescript* y ejecutará el *Javascript* como lenguaje original (Caceres, s.f.).

Se utilizó este lenguaje de programación ya que además de su compatibilidad con Angular posibilitó el mejor y fácil mantenimiento del sistema web y aplicación móvil cuando fueron expuesto a cambios significativos evitando así confusión y sobrecarga.

Visual Studio Code

Editor de código fuente, ligero y potente. Viene con soporte incorporado para *Javascript*, *Typescript* y *Node.js* además consta de un amplio ecosistema de extensiones para otros idiomas y tiempos de ejecución (*Visual Studio Code*, s.f.).

Debido a que el sistema web y aplicación móvil fueron codificados utilizando *Typescript* se utilizó esta herramienta debido a que posee soporte para este lenguaje además de que permitió ejecutar ambos sistemas ya que viene con un terminal incorporado.

2.6.3. Herramientas de Desarrollo Móvil

Ionic

Ionic Framework es un kit de herramientas de *UI* de código abierto para crear aplicaciones móviles y de escritorio de alta calidad y rendimiento utilizando tecnologías *web* (*Html*, *Css* y *Javascript*) (*Ionic*, s.f.).

Esta herramienta ayudó a construir de forma fácil y rápida una amigable interfaz de usuario para la aplicación móvil.

⁵ **Superset:** Se trata de un lenguaje escrito sobre otro lenguaje. En este caso *Typescript* es eso. Obtenido de <https://devcode.la/blog/que-es-typescript/>.

Ionic DevApp

Es una aplicación gratuita que facilita la ejecución de una aplicación *Ionic* directamente en un dispositivo *iOS* o *Android*, evitando problemas de instalación de SDK⁶ nativos (*Ionic*, s.f.).

Con esta herramienta se logró ejecutar la aplicación móvil y a la vez poder visualizar las diferentes pantallas de la misma a través de un *smarthphone* sin necesidad de generar un *APK*⁷.

Apache Cordova

Es un marco de desarrollo móvil de código abierto, que permite utilizar tecnologías *HTML5*, *CSS* y *JS*. Se basan en enlaces *API*⁸ que cumplen con los estándares para acceder a las capacidades de cada dispositivo (Apache Cordova, s.f.).

Con esta herramienta fue posible instalar los plugins compatibles de *Facebook* y *Gmail* para el inicio de sesión, además del plugin necesario para acceder a la galería de fotos del *smarthphone* y modificar la foto en el perfil de usuario.

⁶ **SDK**: es el acrónimo de “*Software Development Kit*” (Kit de desarrollo de software). El *SDK* reúne un grupo de herramientas que permiten la programación de aplicaciones móviles. Obtenido de <https://www.atinternet.com/es/glosario/sdk/>.

⁷ **APK**: *Android Application Package*. Es un archivo ejecutable de aplicaciones para Android. Obtenido de <https://www.samsung.com/latin/support/mobile-devices/what-is-an-apk/>.

⁸ **API**: Interfaz de Programación de Aplicaciones, conjunto de funciones y procedimientos que cumplen una o muchas funciones con el fin de ser utilizados por otro *software*. Obtenido de <https://hipertextual.com/archivo/2014/05/que-es-api/>.

3. RESULTADOS Y DISCUSIÓN

Restricciones de desarrollo

- Se necesita un ordenador con un navegador *web* con conexión a internet para poder manipular tanto el sistema *web* como el móvil.
- El sistema *web* debe tener credenciales de acceso que autentifique al o los administradores ya que es de uso exclusivo para ellos.
- El *Smartphone* en donde se instale *FitApp* debe tener acceso a internet.

Consideraciones

- El super administrador del sistema debe tener conocimientos necesarios para manejar la consola de *Firebase*, además de poder realizar *CRUD*⁹ y gestión de la página principal en el sistema *web*.
- El administrador debe poder realizar *CRUD* de deportistas, recetas, entrenamientos, promociones y categorías.
- El usuario debe tener conocimientos básicos para utilizar un dispositivo inteligente.
- El *Smartphone* debe contar con acceso a internet para poder tener una mejor experiencia de usuario.

Actividades

Historias de Usuario

Las historias de usuario del presente proyecto están disponibles en el Anexo E y Anexo F.

Resumen de historias de usuario

Tabla 3.1 Resumen de historias de usuario

Nº	Historias de usuario	Anexo
1	Inicio de sesión	Anexo E
Historias de usuario del super administrador		
2	Modificar el <i>one page</i>	Anexo E
Historias de usuario del administrador		
3	Modificar perfil	
4	Gestionar categorías de recetas y ejercicios	Anexo E
5	Registrar usuario deportista	

⁹ **CRUD**: Hace referencia a las funciones básicas en las bases de datos. (*Create Read Update and Delete*). Obtenido de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/crud-las-principales-operaciones-de-bases-de-datos/>.

Historias de usuario del administrador	
6	Modificar usuario deportista
7	Eliminar cuenta usuario deportista
8	Registrar recetas
9	Modificar información de recetas
10	Eliminar recetas
11	Registrar ejercicios
12	Modificar información de ejercicios
13	Eliminar ejercicios
14	Registrar rutinas
15	Modificar información de rutinas
16	Eliminar rutinas
17	Registrar entrenamientos
18	Modificar información de entrenamientos
19	Eliminar entrenamientos
20	Visualizar avance de deportistas accediendo a sus perfiles.
Historias de usuario del deportista parte de LSC y externo	
21	Registrar nuevo deportista
22	Iniciar sesión
23	Modificar información de perfil
24	Visualizar recetas
25	Visualizar ejercicios
26	Visualizar entrenamientos
27	Visualizar avance mediante estadísticas.
28	Calcular IMC, peso ideal, calorías diarias y calorías quemadas.

Anexo E

Anexo F

Resumen de *Sprints*

Tabla 3.2 Resumen de Sprints

N° Sprint	Título
<i>Sprint 1</i>	Autenticación
<i>Sprint 2</i>	Súper Administrador
<i>Sprint 3</i>	Entrenador
<i>Sprint 4</i>	Deportista
<i>Sprint 5</i>	Recetas
<i>Sprint 6</i>	Ejercicios
<i>Sprint 7</i>	Entrenamientos
<i>Sprint 8</i>	Rutinas
<i>Sprint 9</i>	Estadísticas
<i>Sprint 10</i>	Registro y <i>Login</i>
<i>Sprint 11</i>	Perfil del Deportista
<i>Sprint 12</i>	Consumo de datos para visualización en app móvil
<i>Sprint 13</i>	Calculadora de calorías

Sprint 0

Objetivo:

- Creación de la base de datos en *Firebase*.
- Activación de los métodos de acceso en *Firebase Authentication*.

Observación Sprint: el sistema presente utiliza una base de datos no relacional *Firebase Real Database* y *Firebase Authentication* ambas utilizadas para el almacenamiento de información, la primera dedicada para los datos correspondientes al *CRUD* del sistema *web* mientras que la segunda dedicada a la autenticación de usuarios en la aplicación móvil y sistema *web*.

Firebase Realtime Database

Firebase permite almacenar la información de modo que ésta tenga una jerarquía que a la vez se va convirtiendo en nodos *JSON*¹⁰ relacionados a una clave aleatoria. Ver Figura 3.1.

Firebase Authentication

Para el almacenamiento de usuarios autenticados en el sistema *web* y aplicación móvil se usó *Firebase Authentication* debido a la necesidad de identificarlos, por tanto, esta plataforma permite que los datos de los usuarios se guarden en la nube de forma segura y personalizada.

Producto Sprint: en la Figura 3.1 se muestra la estructura de la base de datos del sistema que se realizó en *Firebase*.

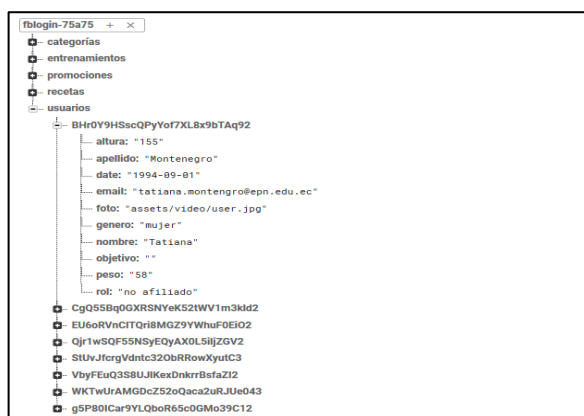


Figura 3.1 Esquema de la base de datos del sistema

¹⁰ **JSON:** *JavaScript Object Notation – Notación de objetos de Javascript* es un formato de texto sencillo para el intercambio de datos. Obtenido de <https://es.wikipedia.org/wiki/JSON>.

en la Figura 3.2 se puede apreciar la estructura de la tabla una vez que los usuarios se hayan autenticado mediante sus credenciales.

Identificador	Proveedores	Creado	Accediste a tu cuenta	UID de usuario ↑
tatiana.montengro@epn.edu...	✉	20 oct. 2019	20 oct. 2019	BHr0Y9HSscQPyyor7XL8x9bTAq92
marco_sanabria1995@hotmail...	f	7 oct. 2019	6 nov. 2019	CgQ55Bq0GXRSNYeK52tWV1m3kl...
rbastidas87@hotmail.com	f	23 oct. 2019	23 oct. 2019	EU6oRVnCiTQi8MGZ9YWhuF0EI02
maybethcsb_010994@hotmail...	G	11 oct. 2019	19 oct. 2019	Qjr1wSQF55NSyEQyAX0LSijZGv2
emidaysan2002@hotmail.com	f	7 oct. 2019	6 nov. 2019	StUvJfcrgVdntc32ObRRowXyutC3
jeandi@gmail.com	✉	19 oct. 2019	10 nov. 2019	VbyFEuQ3S8UJIKexDnkrBsfazI2
jahir@hotmail.com	✉	6 nov. 2019	6 nov. 2019	WKTWUrAMGdcZ52oQaca2uRJUe...
emilyaltamirano2002@gmail...	G	6 nov. 2019	6 nov. 2019	g5P80iCar9YLQboR6Sc0GMO39C12

Figura 3.2 Tabla en *Firebase* de usuarios autenticados

Firebase se encarga de generar el *hash*¹¹ de contraseña brindando seguridad en la información de inicio de sesión del usuario.

3.1. Desarrollo Web

Sprint 1

Objetivo:

- Implementar la autenticación del usuario súper administrador y entrenador

Observación *Sprint*: en este *Sprint* se realizó el diseño del *login* del sistema *web*, validación de acceso y verificación de *email* y contraseña del usuario súper administrador y entrenador.

Producto *Sprint*: Ver Anexo G – *Sprint 1*

Sprint 2

Objetivo:

- Realizar la modificación del *one page*.

Observación *Sprint*: en este *Sprint* se implementó las secciones (inicio, acerca de, servicios, portafolio, contacto) para la modificación del contenido respectivo de cada uno de los apartados en tiempo real dentro del *one page*.

Producto *Sprint*: Ver Anexo G – *Sprint 2*

¹¹ **Hash:** son funciones que, utilizando un algoritmo matemático, transforman un conjunto de datos en un código alfanumérico con una longitud fija. Obtenido de: <https://www.cysae.com/funciones-hash-cadena-bloques-blockchain/>

Sprint 3

Objetivo:

- Realizar la modificación del entrenador.
- Realizar la gestión de categorías de recetas y ejercicios.

Observación Sprint: en este *Sprint* se implementó la capacidad de mediante el acceso a su perfil de usuario en el sistema *web* pueda modificar su información personal accediendo en la sección Mi Perfil y también la sección categorías que le permitirá realizar el *CRUD* de categorías para recetas y ejercicios de esta forma los datos registrados aquí se verán reflejados en el campo tipo de recetas y ejercicios al momento de realizar un registro de cualquiera de estos dos. Todas las acciones mencionadas son realizadas solamente por el entrenador (usuario administrador).

Producto Sprint: Ver Anexo G – *Sprint 3*

Sprint 4

Objetivo:

- Realizar el registro, modificación y eliminación del usuario deportista.
- Visualización del deportista registrado en el sistema *web*.

Observación Sprint: en este *Sprint* se implementó las operaciones *CRUD* registro, modificación y eliminación para el usuario deportista parte de *LSC* y sus validaciones respectivas, cabe mencionar que los campos imagen y rol no son registrados por el usuario entrenador ya que ambos tienen valores establecidos y no aparecen durante el registro. El campo imagen puede ser modificado por el usuario deportista al momento de acceder a su cuenta en la aplicación móvil mientras que el campo rol ayuda a identificar el tipo de usuario deportista, es decir, durante el registro en el sistema *web* este campo está deshabilitado pero activo lo cual indica que es un usuario deportista parte de *LSC*.

Producto Sprint: Ver Anexo G – *Sprint 4*

Sprint 5

Objetivo:

- Implementar el registro, modificación y eliminación de recetas
- Visualizar las recetas registradas en el sistema *web*

Observación *Sprint*: en este *Sprint* se implementó las operaciones *CRUD* de recetas, además de las validaciones respectivas en cada una estas operaciones. Al igual que en el registro de deportistas se puede visualizar las recetas registradas por el usuario entrenador en una tabla en el sistema *web* que permitió ver la variedad de recetas de acuerdo a su tipo. El campo tipo de receta es importante ya que ayudará a filtrar las recetas que visualizará el deportista en la aplicación móvil de acuerdo a su objetivo registrado.

Producto *Sprint*: Ver Anexo G – *Sprint 5*

Sprint 6

Objetivo:

- Implementar el registro, modificación y eliminación de ejercicios
- Visualizar los ejercicios registrados en el sistema *web*

Observación *Sprint*: en este *Sprint* se implementó las operaciones *CRUD* de ejercicios, además de las validaciones respectivas en cada una estas operaciones. También se puede visualizar los ejercicios registrados por el usuario entrenador en una tabla en el sistema *web* que permitirá ver la variedad de ejercicios de acuerdo a su tipo.

Producto *Sprint*: Ver Anexo G – *Sprint 6*

Sprint 7

Objetivo:

- Implementar el registro, modificación y eliminación de rutinas.
- Visualizar las promociones registrados en el sistema *web*

Observación *Sprint*: en este *Sprint* se implementó las operaciones *CRUD* de rutinas, además de las validaciones respectivas en cada una estas operaciones.

También se puede visualizar las rutinas registradas por el usuario entrenador en una tabla en el sistema *web*. Las rutinas están compuestas por ejercicios y solamente estarán disponibles para usuarios deportistas identificados como parte de *LSC*.

Producto *Sprint*: Ver Anexo G – *Sprint 7*

Sprint 8

Objetivo:

- Implementar el registro, modificación y eliminación de entrenamientos
- Visualizar los entrenamientos registrados en el sistema *web*

Observación *Sprint*: en este *Sprint* se implementó las operaciones *CRUD* de entrenamientos, además de las validaciones respectivas en cada una estas operaciones. También se puede visualizar los entrenamientos registrados por el usuario entrenador en una tabla en el sistema *web* que permitirá ver la rutina de ejercicio para un determinado deportista parte de *LSC* y el progreso en porcentaje que este va teniendo en su entrenamiento.

Producto *Sprint*: Ver Anexo G – *Sprint 8*

Sprint 9

Objetivo:

- Visualizar avance de los usuarios deportistas parte de *LSC* accediendo a sus perfiles.

Observación *Sprint*: en este *Sprint* se implementó la visualización del perfil deportista en la tabla deportistas de *LSC* del sistema *web*. Al acceder a uno de los perfiles de un determinado deportista se puede ver toda la información personal del deportista registrado. Los datos que aquí se visualizan son obtenidos de los datos ingresados durante el registro de deportista, siendo los más importantes el peso, altura, fecha de nacimiento fundamentales para poder realizar los cálculos que permitieron obtener valores respecto al peso, IMC en base a los cuáles se logró realizar las gráficas estadísticas.

Producto *Sprint*: Ver Anexo G – *Sprint* 9

3.2. Desarrollo Móvil

***Sprint* 10**

Objetivo:

- Implementar el registro del usuario deportista en la aplicación móvil
- Implementar la autenticación del usuario deportista

Observación *Sprint*: la aplicación móvil permite identificar al deportista, en este caso se tiene dos tipos: el deportista miembro de *LSC* y el deportista externo. De esta manera de acuerdo al tipo de deportista se podrá visualizar y acceder a recursos determinados en la aplicación móvil.

En este *Sprint* se diseñó las ventanas para el registro de un usuario deportista externo, así como también el *login* de la aplicación móvil el cual cuenta con tres formas de autenticarse que son:

- *Email* y contraseña

El deportista miembro de *LSC* puede iniciar sesión solamente con las credenciales proporcionadas por su entrenador personal posterior a su registro en el sistema *web*. En el primer inicio de sesión el usuario y contraseña será el correo electrónico del deportista.

El deportista externo puede también iniciar sesión mediante credenciales obtenidas cuando haya realizado el registro en el formulario disponible en la aplicación móvil.

- *Facebook*

El deportista externo puede acceder a la aplicación *web* usando las credenciales de su cuenta en *Facebook*, para ello fue necesario primero crear una aplicación en *Facebook Developer* y enlazarla con *Firebase*. De la *app* en *Facebook Developer* se obtiene un *APP_ID* y un *APP_NAME* valores necesarios para instalar el *plugin* de Cordova de *Facebook* en el proyecto en *Ionic*.

La documentación de *Ionic* provee de dos comandos utilizados para instalar el plugin de Facebook.

```
$ ionic cordova plugin add cordova-plugin-facebook4 --variable APP_ID="123456789"
--variable APP_NAME="myApplication"

$npm install @ionic-native/facebook
```

- *Gmail*

El deportista externo puede acceder a la aplicación *web* usando las credenciales de su cuenta en *Gmail*. Para ello fue necesario activar el método de acceso de *gmail* en la consola de *Firebase* para obtener los valores de *CLIENT_ID* y *WEB_APPLICATION_CLIENT_ID* necesarios para instalar el *plugin* de Cordova de *Gmail* en el proyecto *Ionic*.

```
$ ionic cordova plugin add cordova-plugin-googleplus --save --variable
REVERSED_CLIENT_ID=myreversedclientid --variable
WEB_APPLICATION_CLIENT_ID=mywebapplicationclientid

$ npm install @ionic-native/google-plus
```

El valor de la variable *REVERSED_CLIENT_ID* debemos remplazarlo por el valor inverso del *ID_CLIENT*. Ejemplo: '123-abc-123.apps.googleusercontent.com' se invierte a com. 'googleusercontent.apps.123-abc-123'

Posteriormente se procedió a realizar la validación de acceso y verificación de *email* y contraseña tanto del usuario deportista externo como del usuario deportista parte de *LSC*.

Producto *Sprint*: Ver Anexo G – *Sprint* 10

***Sprint* 11**

Objetivo:

- Implementar la modificación del perfil de usuario deportista.
- Visualizar los cambios realizados en el perfil de usuario

Observación *Sprint*: en este *Sprint* se implementó la modificación de la información personal presentada en el perfil de usuario de la aplicación móvil. El

deportista puede realizar cambios en su nombre, apellido, peso, altura, fecha de nacimiento, entre otros datos al acceder a su perfil de usuario, al momento de la modificación también se implementó las validaciones correspondientes. Posterior a la edición, los datos cambiados se verán reflejados en el perfil de usuario tanto de la aplicación móvil como del sistema *web* en caso de que el entrenador acceda al perfil de un determinado deportista *LSC* en el sistema *web*.

Producto Sprint: Ver Anexo G – Sprint 11

Sprint 12

Objetivo:

- Implementar la visualización los datos de recetas, ejercicios, entrenamientos consumidos desde *Firebase*.
- Implementar la visualización del avance del usuario deportista mediante gráficos estadísticos.

Observación Sprint: en este *Sprint* se realizó la implementación del consumo de los datos de recetas, ejercicios, entrenamientos registrados en el sistema *web*. El servidor *Firebase* permitió acceder a estos datos y poder reflejarlos en la aplicación móvil, de manera que el usuario deportista pueda acceder a estos recursos. Cabe mencionar que las promociones serán recibidas como notificación sólo cuando el usuario se autentique como deportista parte de *LSC*.

Producto Sprint: Ver Anexo G – Sprint 12

Sprint 13

Objetivo:

- Implementar el cálculo de calorías (IMC, peso idea, calorías diarias, calorías quemadas)
- Implementar la visualización de los resultados obtenidos luego del cálculo de calorías

Observación Sprint: en este *Sprint* se realizó la implementación del cálculo de calorías, con este podrán saber calcular su IMC (índice de masa corporal), calcular su peso ideal, calcular las calorías diarias a consumir y también calcular las calorías

quemadas tras realizar un deporte ya sea caminata, correr, natación o ciclismo, entre otras, para ello cada uno cuenta con un formulario disponible en el menú entrenamiento en la sección calculadora. Cada formulario pide información específica del peso, altura, edad, género que la obtenemos de los valores ingresados durante el registro del deportista.

Durante el cálculo los datos ingresados por el usuario son validados para la correcta ejecución del cálculo y posteriormente una vez realizado el cálculo, los datos obtenidos son visualizados por el usuario deportista.

Producto *Sprint*: Ver Anexo G – *Sprint* 12

3.3. Pruebas de Funcionamiento

Pruebas del Sistema *Web*

Pruebas de caja negra

Las pruebas de caja negra detalladas en la Tabla 3.3 fueron realizadas con el objetivo de comprobar que el *software* realice lo que el cliente solicitó. Fueron basadas en los requerimientos funcionales sin necesidad de conocer su codificación.

Tabla 3.3 Pruebas de Caja Negra en el Sistema *Web*

Id	Descripción	Estado	Observación	Anexo
RFW2: Modificación del <i>One Page</i>				
PCN1	El super administrador debe iniciar sesión para poder visualizar todo el menú que le corresponder a su cuenta y posteriormente le permitirá cerrar sesión.	Aprobado	En esta prueba se implementó un mensaje de alerta si el usuario como algún error al ingresar las credenciales para el ingreso como super administrador.	Anexo H

Id	Descripción	Estado	Observación	Anexo
RFW2: Modificación del One Page				
PCN2	El super administrador puede gestionar el contenido de la página <i>web</i> principal, al iniciar sesión podrá ingresar a las secciones (inicio, acerca de mí, servicios, portafolio, contacto) y poder actualizar el contenido de acuerdo a las necesidades de <i>LSC</i>	Aprobado	Al momento de gestionar contenido en cualquiera de las secciones mencionadas en la descripción , en caso de añadir, modificar o eliminar la información de los formularios dispuestos en cada sección el sistema desplegará los mensajes de alerta apropiados al caso.	Anexo I
RFW5: Modificación del Perfil Administrador				
PCN3	El administrador debe iniciar sesión para poder visualizar todo el menú que le corresponde a su cuenta y posteriormente le permitirá cerrar sesión	Aprobado	En este prueba se implementó un mensaje de alerta si el usuario como algún error al ingresar las credenciales para el ingreso como super administrador.	Ver Anexo J
RFW6: Registro, modificación y eliminación de usuario deportista RFW7: Registro, modificación y eliminación de recetas RFW8: Registro, modificación y eliminación de ejercicios RFW9: Registro, modificación y eliminación de rutinas RFW10: Registro, modificación y eliminación de entrenamientos				
PCN4	El administrador puede realizar operaciones <i>CRUD</i> de deportistas , recetas, ejercicios, entrenamientos y rutinas además de poder modificar su perfil de usuario.	Aprobado	Para acceder a los formularios adecuados debe seguir las secciones disponibles en el menú izquierdo de la pantalla si desea añadir, modificar o eliminar la información de los formularios dispuestos en cada sección el sistema desplegará los mensajes de alerta apropiados al caso.	Ver anexo K

Id	Descripción	Estado	Observación	Anexo
RFW11: Visualizar avance del usuario deportista mediante gráficos estadísticos				
PCN5	El administrador puede dar seguimiento al progreso de sus deportistas mediante gráficos estadísticos.	Aprobado	En esta prueba fue indispensable manejar los datos obtenidos en el registro de deportistas como peso, altura para poder obtener valores de IMC que son representados gráficamente para que el administrados pueda realizar un seguimiento.	Ver anexo L

Pruebas de compatibilidad

Las pruebas de compatibilidad del sistema *web* se realizaron en diferentes navegadores como se muestra en la Figura 3.3 y en la Tabla 3.4 con el fin de visualizar el correcto funcionamiento del sistema *web*.

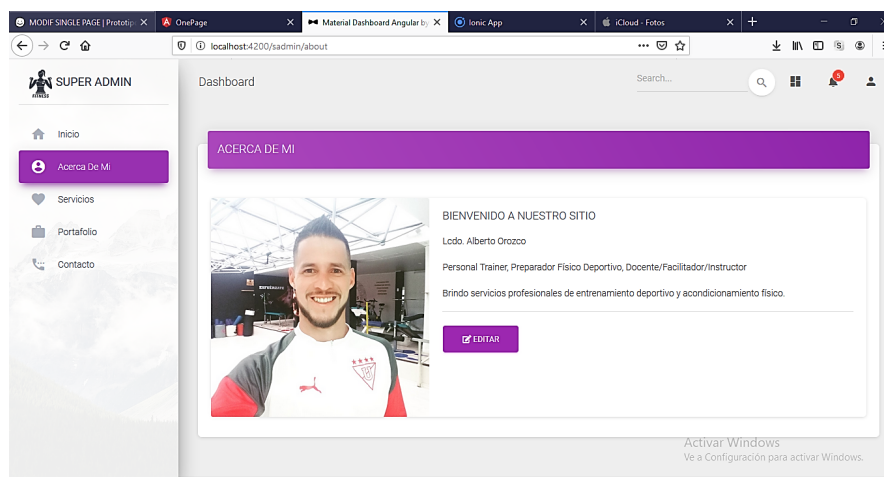


Figura 3.3 Sistema *Web* visto desde el navegador *Firefox*

Tabla 3.4 Navegadores que soporta el Sistema *FitApp*

Plataforma	Versión
<i>Google Chrome</i>	77.0.3865.120
<i>Mozilla Firefox</i>	69.0.3
<i>Microsoft Edge</i>	44.18362.387.0

Pruebas de la Aplicación Móvil

Pruebas de compatibilidad

Las pruebas de compatibilidad se realizaron con el objetivo observar cómo reacciona la aplicación móvil en diferentes *Smartphone* con sus respectivos sistemas operativos y versiones. Para realizar las pruebas en *Android* se utilizó el comando `ionic serve -l` que ejecutará el navegador con una la dirección `http://localhost:8200/` en donde se puede elegir la plataforma a utilizar para la ejecución.

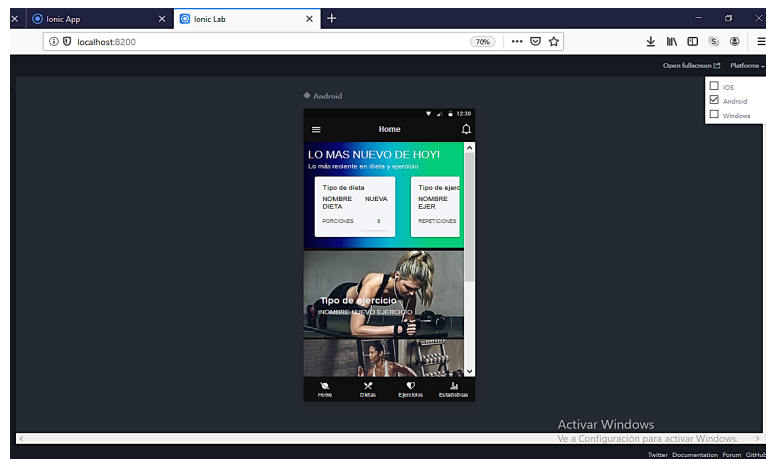


Figura 3.4 Visualización de la app móvil en ionic lab

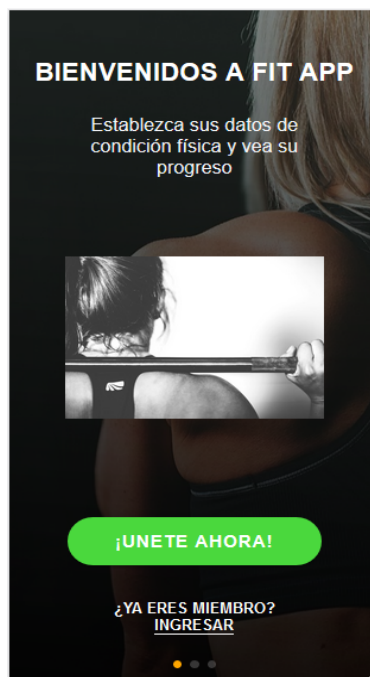


Figura 3.5 Visualización en plataforma Android ionic lab

Mientras que las pruebas en *iOS* se realizaron en *DevApp*, el cual se descarga de *AppStore*, para ello debemos ejecutar el proyecto utilizando el siguiente comando *ionic serve --devapp* lo que devolverá el nombre de nuestra app que podremos visualizar en la aplicación *DevApp* como muestra la Figura 3.6 y Figura 3.7 o podemos ingresar manualmente los datos. Ver Tabla 3.5.

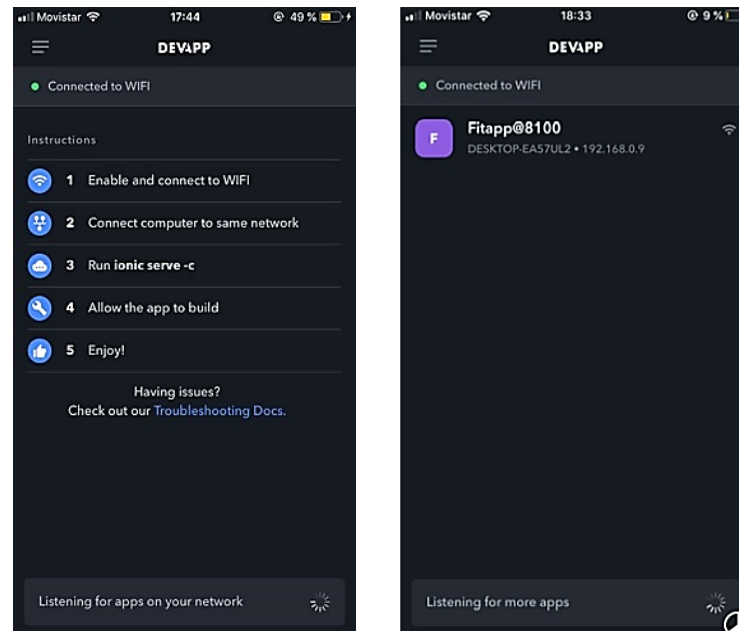


Figura 3.6 Vista del proyecto *FitApp* en *DevApp*

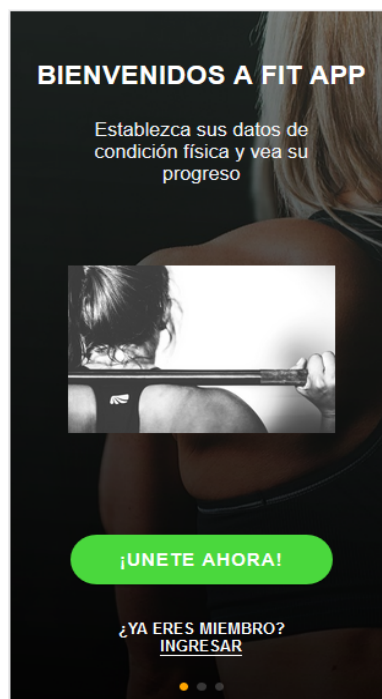


Figura 3.7 Pantalla de la aplicación *FitApp* en *Ionic DevApp*

Tabla 3.5 Pruebas de compatibilidad en *Smartphones*

Plataforma	Versión	Modelo	Anexo
<i>Android</i>	7.0	<i>Samsung J5 Prime</i>	Anexo M
<i>Android</i>	8.0.0	<i>Sony Xperia</i>	Anexo N
<i>iOS</i>	12.4.1, 13.1.3	<i>Iphone 6s</i>	Anexo Ñ

Pruebas de interfaz con el usuario

Con estas pruebas se evalúa los siguientes aspectos:

- **Elementos gráficos:** orden de la pantalla, colores y contrastes, fondos, imágenes adecuadas. Ver Figura 3.8.
- **Tipografías y textos:** ortografía, redacción y símbolos. Ver Figura 3.9.
- **Entradas:** datos de entrada, acciones posteriores, adecuada posición y tamaño. Ver Figura 3.10 y Figura 3.11.
- **Mensajes:** de error y alerta. Ver Figura 3.12 y Figura 3.13.

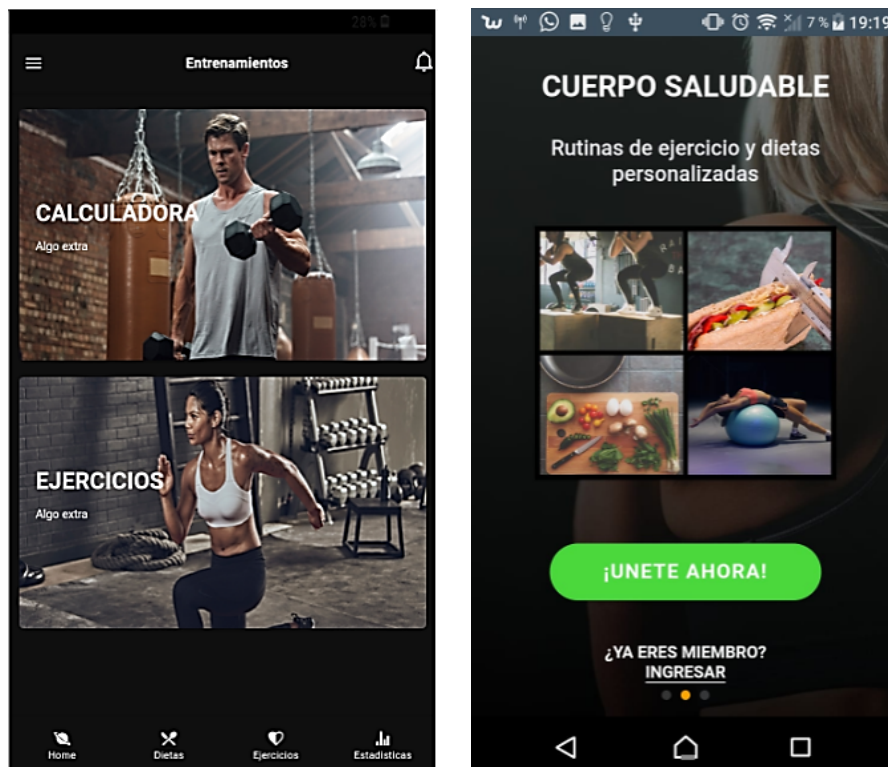


Figura 3.8 Prueba de interfaz: elementos gráficos

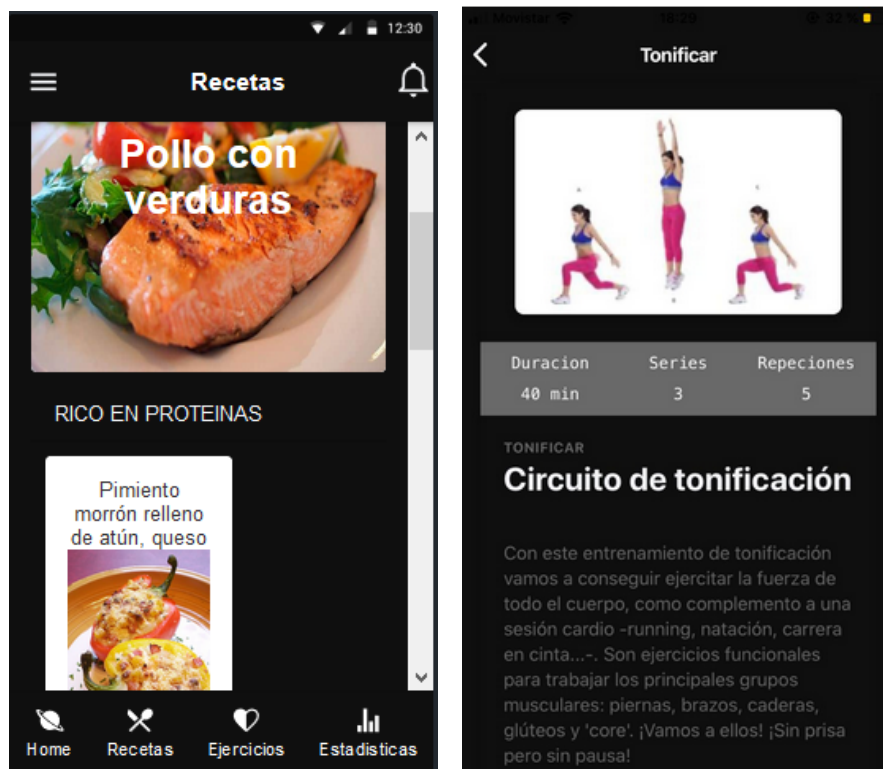


Figura 3.9 Prueba de interfaz: tipografías y textos

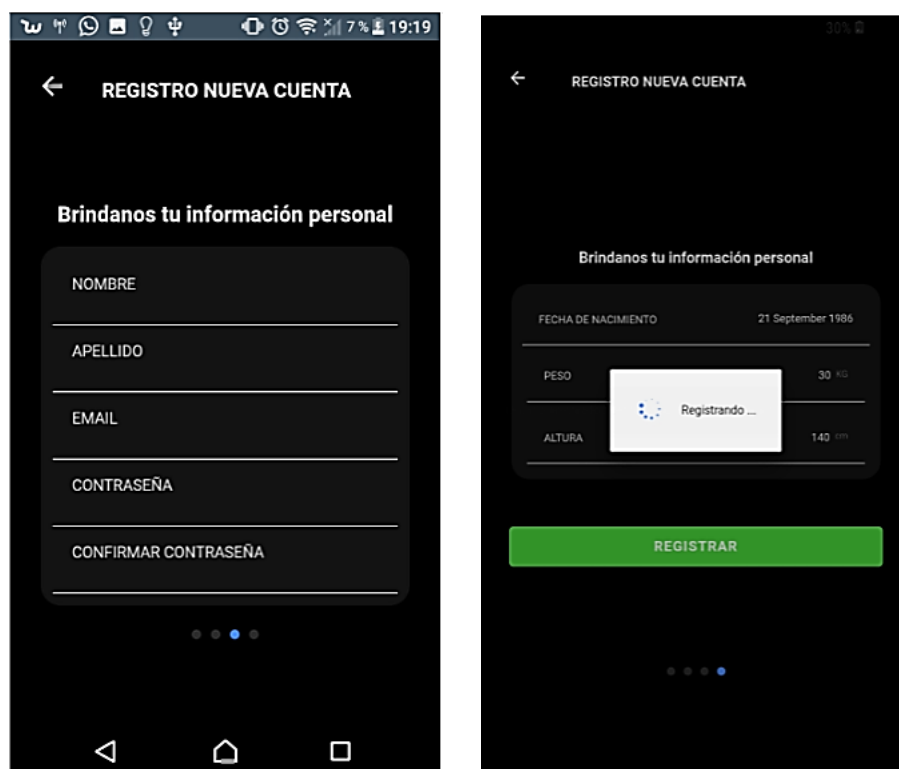


Figura 3.10 Datos de entrada y acciones al registrarlo

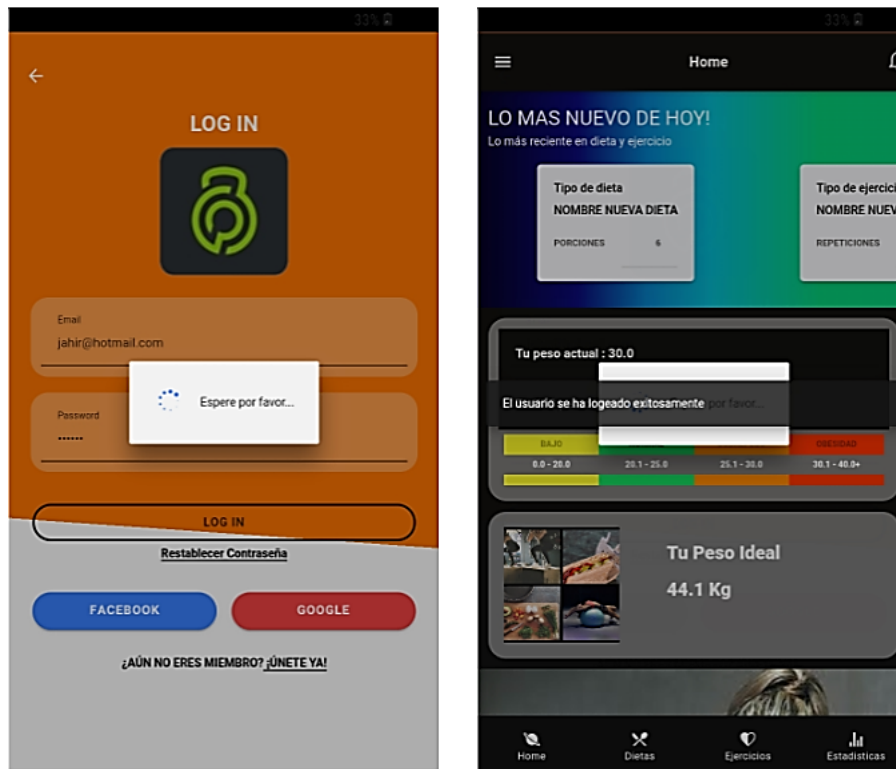


Figura 3.11 Datos de entrada para iniciar sesión y acciones al iniciar

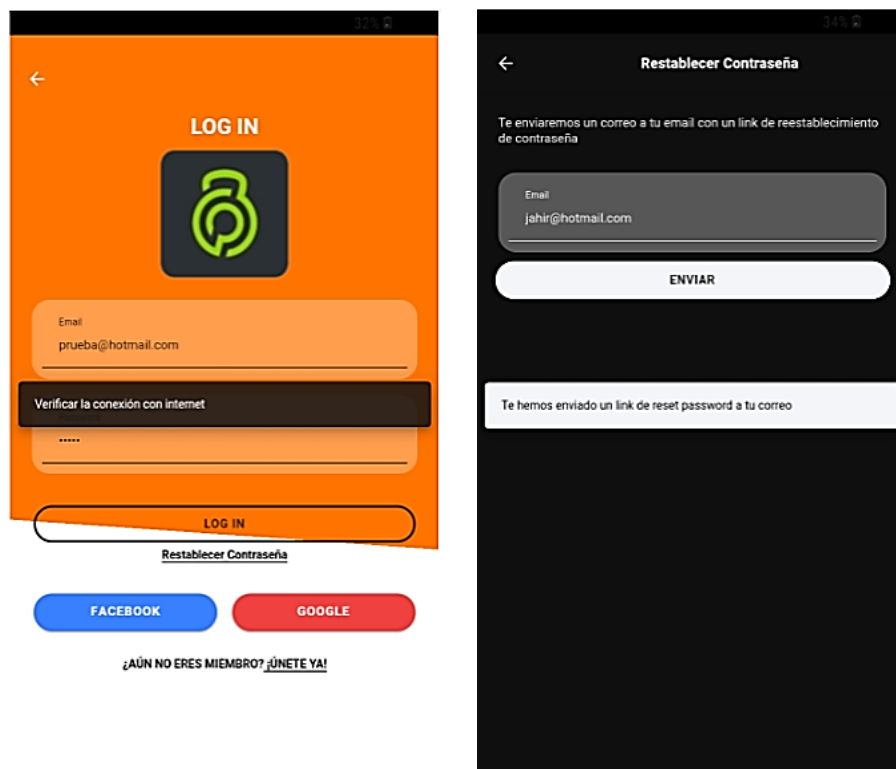


Figura 3.12 Mensajes de alerta en la aplicación móvil

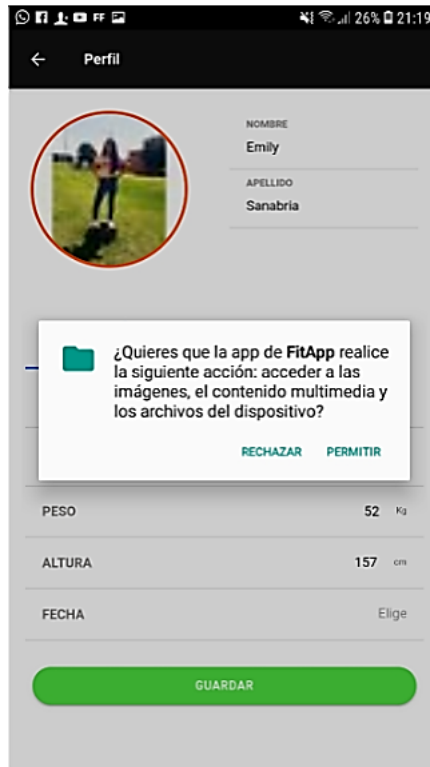


Figura 3.13 Mensaje al desear ingresar a la galería de fotos del dispositivo

4. CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Los requerimientos estudiados permitieron agilizar el proceso de desarrollo del sistema al brindar información exacta acerca de los procesos a automatizar en este trabajo.

La metodología *Scrum* permitió al equipo de desarrollo concentrarse en funcionalidades puntuales por intervalos de tiempo, logrando mejores entregables ligados a las necesidades del *Product Owner* cumpliendo con requerimientos establecidos o nuevos y permitiendo una constante retroalimentación con el *Development Team* para poder adaptar el desarrollo a las cambiantes necesidades del sistema.

La implementación de *Firebase* y su servicio en la nube *Cloud Storage* permitió guardar datos y a la vez recuperarlos como objetos *JSON* empleando lapsos cortos de tiempo y conservando su integridad. Utilizar esta estructura facilitó la lectura e interpretación de los resultados obtenidos. *Firebase* y *Cloud Storage*

también proporcionan beneficios como una fácil configuración y el consumo eficiente de la información almacenada mostrando cambios al instante.

Para la autenticación se utilizó *Firebase Authentication* ya que proporciona sus propios métodos para la autenticación de usuarios, permitiendo que los usuarios ingresen al sistema *web* garantizando la protección de sus datos.

Se utilizó *Balsamiq Cloud* como herramienta gráfica que permitió diseñar bocetos de las interfaces de usuario del sistema *web* y de la aplicación móvil de una manera fácil y ágil. Este proceso de prototipado ayudó a crear el sistema lo más rápido y parecido posible a los bocetos ya creados evitando el consumo de más tiempo en el diseño de las interfaces de usuario.

Para el desarrollo del sistema *web* se utilizó Angular ya que al ser una herramienta con una curva de aprendizaje sencilla permitió reducir tiempos y facilitó la creación del *single page* del sistema *web*.

Para el desarrollo de la aplicación móvil se utilizó el *framework open source Ionic*, ya que permitió el desarrollo de aplicaciones híbridas además de proveer componentes necesarios para crear una interfaz atractiva y fácil de manejar por los usuarios.

Se realizaron diferentes pruebas con el fin de comprobar el funcionamiento, compatibilidad e interfaz con el usuario. Las pruebas de *software* realizadas al sistema *web* estuvieron basadas en los requerimientos funcionales del mismo y permitieron verificar el cumplimiento de los mismos de acuerdo a lo que el cliente solicitó.

Las pruebas de compatibilidad se realizaron con el fin de verificar el correcto funcionamiento en diferentes navegadores *web* y plataformas y a través de las mismas se pudo establecer la versión adecuada para el funcionamiento del sistema *web*, mientras que para la aplicación móvil permitieron comprobar el correcto funcionamiento en distintos dispositivos móviles y a través de las mismas se pudo delimitar su versión mínima.

Finalmente, las pruebas de interfaz con el usuario permitieron verificar que tan amigables e intuitivas son al momento del uso, por ejemplo, si el usuario logró iniciar sesión o modificar su perfil. Además, a través de estas se logró identificar posibles errores para poder corregirlos en posteriores versiones de la aplicación.

Recomendaciones

Es necesario tener una conexión a internet para poder tener una mejor experiencia de usuario tanto en la móvil como en la *web* y así poder visualizar los cambios realizados.

Se recomienda trabajar con las versiones estables tanto de Ionic como de Angular ya que ambos son *framework* que se van actualizando constantemente, debido a la compatibilidad de los *plugins* a usar.

5. REFERENCIAS BIBLIOGRÁFICAS

AngularJS. (2018). *docs.angularjs.org*. Recuperado el 8 de Mayo de 2019, de <https://docs.angularjs.org/guide/introduction>

Apache Cordova. (s.f.). *Apache Cordova*. Recuperado el 17 de Octubre de 2019, de <https://cordova.apache.org/docs/en/latest/guide/overview/index.html>

Caceres, M. (s.f.). *DevCode*. Recuperado el 9 de Mayo de 2019, de <https://devcode.la/blog/que-es-typescript/>

El Telégrafo. (30 de Agosto de 2015). *El Telégrafo*. Recuperado el 31 de Octubre de 2019, de <https://www.eltelegrafo.com.ec/noticias/buen/1/la-proporcion-de-personas-que-practican-deporte-aumento-9-puntos-en-ecuador>

Firebase. (s.f.). *Firebase.google.com*. Recuperado el 8 de Mayo de 2019, de <https://firebase.google.com/products/auth/?hl=es-419>

Ionic. (s.f.). *Ionic Framework*. Recuperado el 9 de Mayo de 2019, de <https://ionicframework.com/docs/intro>

- Ionic. (s.f.). *Ionic Framework*. Recuperado el 9 de Mayo de 2019, de <https://ionicframework.com/docs/appflow/devapp>
- Lara, W. (2015). *Platzi*. Recuperado el 22 de Octubre de 2019, de <https://platzi.com/blog/metodologia-scrum-fases/>
- Palacio, J. (2015). *scrummanager.net*. Recuperado el 3 de Mayo de 2019, de https://www.scrummanager.net/files/scrum_l.pdf
- Requena Mesa, A. (18 de Diciembre de 2018). *OpenWebinars*. Recuperado el 3 de Mayo de 2019, de <https://openwebinars.net/blog/roles-scrum/>
- Robles, V. (2017). *victorroblesweb.es*. Recuperado el 8 de Mayo de 2019, de <https://victorroblesweb.es/2017/08/05/que-es-angular-y-para-que-sirve/>
- Schwaber , K., & Sutherland, J. (Noviembre de 2017). *scrumguides.org*. Recuperado el 3 de Mayo de 2019, de <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-SouthAmerican.pdf#zoom=100>
- Tutorials Point. (s.f.). *Tutorials Point*. Recuperado el 17 de Octubre de 2019, de https://www.tutorialspoint.com/angular_material/index.htm
- Villarroel, R. (20 de Enero de 2017). *ingenieriadesoftwareutmachala.wordpress.com*. Recuperado el 12 de Mayo de 2019, de <https://ingenieriadesoftwareutmachala.wordpress.com/2017/01/20/requerimientos-funcionales-y-no-funcionales/>
- Visual Studio Code. (s.f.). *visualstudio.com*. Recuperado el 9 de Mayo de 2019, de <https://code.visualstudio.com/docs>

ANEXOS