



UNIVERSIDAD JOSÉ ANTONIO PÁEZ

**DESARROLLO DE UNA
APLICACIÓN MULTIPLATAFORMA
PARA EL CONTROL DEL
DESEMPEÑO DEL DEPORTISTA**

Autor: Omar Pérez
C.I. 20.227.222

Urb. Yuma II, calle N° 3. Municipio San Diego
Teléfono: (0241) 8714240 (master) – Fax: (0241) 8712394



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA
INGENIERÍA EN COMPUTACIÓN

**DESARROLLO DE UNA APLICACIÓN MULTIPLATAFORMA
PARA EL CONTROL DEL DESEMPEÑO DEL DEPORTISTA**

Trabajo de Grado presentado como requisito parcial para optar al título de
INGENIERO EN COMPUTACIÓN

Autor: Omar Perez
C.I.: V-20.227.222
Tutor: Ing. Mayerlin Maldonado
C.I. V- 11.810.356

San Diego, Julio 2018



Universidad José Antonio Páez
Facultad de Ingeniería

FI-C-015-2018-1

Valencia, 30 de Mayo de 2018

Ciudadano:
Perez Omar
C.I. 20.227.222
Presente.-

Cumplo con informarle que la Comisión de Trabajo de Grado y Pasantías de la Facultad de Ingeniería en su reunión N° 1-2018 de fecha 30/05/2018 aprobó el proyecto de trabajo de grado titulado **DESARROLLO DE UNA APLICACIÓN MULTIPLATAFORMA PARA EL CONTROL DEL DESEMPEÑO DEL DEPORTISTA**. Presentado por usted como requisito para optar al título de Ingeniero en Computación.

Se ratifica la designación de la Ing. Mayerlín Maldonado, C.I. 11.810.356 y la Ing. Alicia Yanez de Pizzella, C.I. 4.598.880 como Tutores Académicos que lo asesorarán en el desarrollo de este proyecto.

Atentamente,

Zalcedo

Prof. Zulay Salcedo
Decana de la Facultad de Ingeniería



c. c. Coordinación de Pasantías y Trabajo de Grado (1).

ZS/gr



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE COMPUTACIÓN

ACEPTACIÓN DEL TUTOR

Quien suscribe, Ing. Mayerlin Maldonado, portador(a) de la cédula de identidad N° 11.810.356, en mi carácter de tutor del trabajo de grado presentado por el(la) ciudadano Omar E. Perez K., portador de la cédula de identidad N° 20.227.222, titulado **DESARROLLO DE UNA APLICACIÓN MULTIPLATAFORMA PARA EL CONTROL DEL DESEMPEÑO DEL DEPORTISTA**, presentado como requisito parcial para optar al título de Ingeniero en Computación, considero que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del jurado examinador que se designe.

En San Diego, a los treinta y un días del mes de Julio del año dos mil dieciocho.


Firma
Ing. Mayerlin Maldonado
V- 11.810.356

AGRADECIMIENTOS

Me encuentro sumamente agradezco con mis padres Jorge Pérez y María Kuzma, por brindarme su apoyo continuo e incansable en todo el camino que he recorrido hasta este momento; por la paciencia, apoyo y motivación que han tenido en mí para la culminación de este trabajo de grado. Igualmente, agradezco enormemente mi familia por el ejemplo de determinación, esfuerzo y dedicación me han dado a lo largo de mi vida.

A mi esposa la cual amo le estoy profundamente agradecido por su comprensión, dedicación y cariño en todo momento de este proceso tan largo y arduo.

A mi tutora Mayerlin Maldonado, por su orientación, comprensión y fe en todo momento, quien me ayudó con su experiencia, consejos y anécdotas para lograr la culminación de este proyecto.

Agradezco especialmente a la comunidad Android y especialistas en las áreas de estudio con las que se trabajaron, por su arduo esfuerzo e incontables horas dedicadas para poder brindar todo el conocimiento valioso, el cual fue de suma utilidad en esta investigación.

Y por último agradezco a la universidad y sus profesores por haberme enseñado tanto en estos años y por ser mi casa de estudio la cual he aprendido tanto y he tenido muy buenos momentos.

OMAR ELIAS, PEREZ KUZMA

DEDICATORIA

A mi madre, por su apoyo y amor incondicional a lo largo de estos años, así como el pilar que ha sido a lo largo de mi vida.

A mi padre por su cariño, sabios consejos y haber sido el tutor de mi formación.

A mi primo José Domador y mi Prima Stela Domador, por ser como unos hermanos para mí, y por haberme enseñado que la vida tiene tantas cosas bellas.

A mis familiares, por brindarme su amor, compañía y apoyo en todo momento.

OMAR ELIAS, PEREZ KUZMA

ÍNDICE GENERAL

RESUMEN.....	ix
INTRODUCCIÓN	1
EL PROBLEMA	3
Planteamiento del problema	3
Formulación del problema.....	5
Objetivos de la investigación.....	5
Justificación de la Investigación.....	6
Alcance	7
MARCO TEÓRICO	8
Antecedentes de la Investigación	8
Bases Teóricas	12
Definición de Términos Básicos	26
MARCO METODOLÓGICO.....	29
Tipo de investigación	29
Diseño de la Investigación.....	29
Nivel de la investigación	30
Población y Muestra	30
Técnicas e Instrumentos de recolección de datos.....	30
Fases metodológicas	31
RESULTADOS.....	33
Fase I: Diagnosticar situación actual del desempeño los deportistas en Valencia estado Carabobo mediante técnicas de observación.....	33
Fase II: Determinar los requerimientos funcionales y no funcionales de la aplicación para el establecimiento de la funcionalidad de la aplicación, aplicando técnicas de recolección de datos.....	37
FASE III: Diseñar una aplicación multiplataforma mediante esquemas de modelos para las distintas partes de la aplicación.....	45

FASE IV: Desarrollar la aplicación multiplataforma a través de las herramientas de programación seleccionadas.....	53
CONCLUSIONES Y RECOMENDACIONES.....	80
5.1 Conclusiones	80
5.2 Recomendaciones	82
REFERENCIAS	83
Bibliografía	83
ANEXOS	¡Error! Marcador no definido.
ANEXO A	86
ANEXO B	88
ANEXO C-1	90
ANEXO C-2	91



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

**DESARROLLO DE UNA APLICACIÓN MULTIPLATAFORMA
PARA EL CONTROL DEL DESEMPEÑO DEL DEPORTISTA**

Autor: Omar Elías Pérez Kuzma
Tutor: Ing. Mayerlin Maldonado
Fecha: Marzo 2018

RESUMEN

El presente trabajo de investigación hace referencia al desarrollo de una aplicación Android y web para el control del desempeño de los deportistas en Valencia estado Carabobo con el fin que los deportistas en especial en el ámbito del futbol puedan tener una herramienta que les facilite la organización de los equipos y partidas, darse a conocer el ámbito, tener un control de sus entrenamientos a través de fórmulas de quema de calorías y alimentación. Esta investigación es de tipo proyecto especial el cual es una propuesta viable para dar solución frente a una problemática, posee un nivel de investigación descriptivo. La metodología de desarrollo utilizada es programación extrema (XP). La investigación dentro del contexto de Gestión de proyectos y tecnologías de información y comunicación.

Descriptor: Aplicación Multiplataforma, Entrenamiento, Futbol, Formulas Calóricas, Deporte, Android.

INTRODUCCIÓN

Desde los inicios de los tiempos el ser humano siempre fue activo físicamente por las condiciones desfavorables en su entorno, ya sea bien, por cazar a sus presas para alimentarse, movilizarse en grandes distancias o escapar de sus depredadores. Hoy en la actualidad encontramos que su entorno ha cambiado favorablemente, haciendo la vida más fácil como la conocemos, ya no es necesario cazar el alimento porque este se encuentra fácilmente en los supermercados, recorrer distancias tampoco es un problema debido a que existen diversos medios de transportes los cuales son rápidos y no requieren de un agotamiento físico. Al punto que se quiere llegar en cuestión es que por la facilidad de la vida que lleva el ser humano ha disminuido su actividad física.

La inactividad física y la mala alimentación son factores que contribuyen a la obesidad, trayendo consigo problemas en la salud. Por este motivo se pensó en el desarrollo de una aplicación la cual cuente con las herramientas necesarias para que las personas que desean cambiar sus malos hábitos puedan hacerlo mediante un control de su quema de calorías y alimentación.

Dentro de lo que es la aplicación en sí, las personas que aspiran hacer deporte o ya lo practican podrán organizarse e interactuar, también contará con un registro de la evolución del entrenamiento físico de la persona para que pueda comparar su estado pasado y actual de entrenamiento, por otra parte las personas que quieran darse a conocer en el ámbito de este deporte podrán hacerlo.

Para esta aplicación se escogió el fútbol como deporte, ya que es un deporte que requiere de gran actividad física tanto en sus entrenamientos como partidos. También el fútbol es un deporte muy visto a nivel mundial y hoy en

día los jóvenes Venezolanos sienten una gran fanaticada por sus ídolos futbolísticos.

La presente investigación está estructurada en cuatro capítulos, que serán descritos a continuación:

Capítulo I: en este capítulo se describe el problema con más detalles, realizando énfasis sobre la problemática existente, factores que influyen en el desarrollo de una aplicación de multiplataforma para el control del deportista y los objetivos a alcanzar con el desarrollo del proyecto, su justificación y alcance.

Capítulo II: se explican las bases teóricas, los fundamentos en los que se basa la investigación, se exponen las definiciones pertinentes al tema y necesarios para el desarrollo de la presente investigación.

Capítulo III: se describe la metodología utilizada, el nivel de la investigación, las fases metodológicas que componen este proyecto, la población y muestra a estudiar, y las técnicas e instrumentos de recolección de datos.

Capítulo IV: se muestran los recursos que serán utilizados en la realización de cada una de las partes que componen el desarrollo de este sistema y finalmente son listadas las referencias bibliográficas utilizadas en la investigación.

EL PROBLEMA

Planteamiento del problema

El avance tecnológico que ha surgido en los últimos años ha conducido a la imperiosa necesidad de llevar a cabo la sistematización de casi todos los procesos que la permiten, esto con el fin de simplificar las actividades que se realizan tanto laborales como personales. A nivel mundial se debe tener presente que con una tecnología de punta los resultados serán los deseados, debido a que cada institución o persona posee distintas responsabilidades y estas ayudan al buen desarrollo, logro, cumplimiento de objetivos y metas planificadas. Siguiendo con el mismo orden de ideas, se puede decir que la tecnología es capaz de acoplarse a los requerimientos inmediatos y futuros de cualquier situación, es por ello que se pensó en el desarrollo de una aplicación multiplataforma para el control del desempeño del deportista.

Partiendo de esto se plantea una problemática, hoy día la actividad física en los adultos jóvenes ha mermado debido a diversos factores como lo son: lo social, económicos y disponibilidad de tiempo, causando esto el aumento de sedentarismo, lo cual trae como consecuencia problemas de salud. Por años realizar actividad física se ha recomendado como una de las principales soluciones médicas para mantener un buen estado de la misma.

En el mismo orden de ideas y haciendo referencia a los factores mencionados, en la sociedad venezolana no se le ha dado la importancia que amerita realizar actividad física, la cual conlleva como beneficio para la población un desarrollo integral. En el factor económico las personas que aspiran realizar algún deporte se les dificulta la práctica del mismo, debido a que practicarlo hoy en día es costoso, de igual manera los medios para desplazarse y puntos de encuentro son escasos, por otra parte la población que

es sustento de familia dispone de poco tiempo, por lo que, la disponibilidad del mismo debe ser eficientemente aprovechado.

Se escogió el Fútbol como deporte para utilizarlo como contexto dentro de la aplicación ya que este deporte requiere de una gran actividad física, ejercitarse por un lapso de 90 minutos activa el sistema cardiovascular, mejora el consumo de oxígeno, ayuda al aumento de masa muscular y también a reducir los niveles de estrés del individuo, trayendo consigo una mejora sustancial en la salud del individuo. Por otra parte, en el transcurso del tiempo hemos visto como el Fútbol ha evolucionado y se ha convertido en parte importante de la sociedad, tanto así que es uno de los deportes más jugado y visto a nivel mundial, sin embargo, en nuestro país no cuenta con las herramientas necesarias para los deportistas que aspiran divertirse, hacer ejercicio o darse a conocer en este ámbito.

A pesar de que el fútbol ha empezado a tener auge en la cultura del venezolano, no existe una tradición como en el caso del béisbol que es el deporte que tiene más aceptación en Venezuela, según la encuesta realizada por IVAD Consultores entre agosto y septiembre de 2011, en la cual indican que al 88,3% de la población venezolana le gusta el béisbol (9 de cada 10 de venezolanos). Además la liga venezolana de béisbol cuenta con un buen desempeño en competencias internacionales dentro de los cuales se encuentran entrenadores y jugadores con experiencia.

Por otra parte tenemos que, gracias a los avances tecnológicos, se cuenta con los medios para que las persona interactúen de una manera fácil y rápida. En Venezuela son muy vistas las diferentes redes sociales como lo son el “Instagram” cuyas funciones son subir fotos y vídeos, pudiendo aplicarles efectos audiovisuales, “Facebook” la cual es una plataforma en la que los usuarios pueden mantenerse en contacto continuo con un grupo de amistades e intercambiar con ellos cualquier tipo de contenido y “Twitter” la cual es una

aplicación gratuita de microblogging que reúne las ventajas de los blogs, las redes sociales y la mensajería instantánea.

Haciendo referencia en lo anterior, no existe ninguna como tal orientada a la interactividad entre los deportistas y que puedan tener un control de su desempeño deportivo, por esas razones se necesitan crear nuevas herramientas y estrategias para las personas que desean practicar algún deporte como el futbol, puedan mejorar su salud, aportarles beneficios a su calidad de vida y darse a conocer en el medio.

Formulación del problema

De acuerdo a lo planteado anteriormente se formula la siguiente interrogante: ¿Cómo se podría optimizar el control del desempeño del jugador de futbol en Valencia estado Carabobo?

Objetivos de la investigación

1.1.1 Objetivo General

Desarrollar una aplicación multiplataforma, para el control del desempeño de los adultos deportistas y así fomentar la práctica del deporte.

1.1.2 Objetivos Específicos

- Diagnosticar situación actual del desempeño los deportistas en el Valencia estado Carabobo mediante técnicas de observación.
- Determinar los requerimientos funcionales y no funcionales para el establecimiento de la funcionalidad de la aplicación, aplicando técnicas de recolección de datos.
- Diseñar una aplicación multiplataforma mediante esquemas de modelos para las distintas partes de la aplicación.
- Desarrollar la aplicación multiplataforma a través de las herramientas de programación seleccionadas.

Justificación de la Investigación

Con la implementación de este software se pretende que las personas de edades comprendidas entre 18 y 45 años puedan practicar el fútbol y así disminuir el sedentarismo, ya que esto aportaría mejoras para su salud y calidad de vida. Además de, proporcionarles un espacio recreativo el cual disminuya los niveles de estrés y ansiedad. También por otra parte se pretende fomentar este deporte en el estado Carabobo, de manera que siendo éste un estado a nivel nacional altamente reconocido por sus jugadores en el ámbito del fútbol, poder ayudar al crecimiento del mismo.

De igual manera esta implementación también permite reclutar jugadores para los equipos o clubes mediante el reconocimiento adquirido dentro de la aplicación, ver el rendimiento propio de cada jugador a través de estadísticas, facilitará a los interesados en la práctica de este deporte ponerse en contacto mediante la interacción dentro de la aplicación y desarrollar distintas actividades propias del deporte como lo son los encuentros amistosos y torneos.

En el desarrollo de esta aplicación se usaron las nuevas tecnologías como lo son la geo localización satelital que le permitirá encontrar a los equipos contrincantes para practicar el deporte de manera precisa en el mapa de Google, encriptado para los datos sensibles los cuales son necesarios para tener un cierto nivel de seguridad, estadísticas personalizadas y transmisión de información mediante código “QR” que aportara la reducción de tiempos en los procesos de organización de la partida, invitación de amistadas y resultados .

Alcance

Esta aplicación fue subida en la Web y la Play-Store de Google, pretende ser gratuita para su fácil descargar y posteriormente promocionarla en el estado Carabobo, principalmente en las entidades, canchas, complejos los cuales tengan un espacio para la práctica del fútbol. Por otra parte, dentro de las funcionalidades planteadas, se tiene que la aplicación cuenta con un sistema de login de usuarios a través de Facebook o de la misma aplicación, el cual tendrá sesiones a través de “Tokens” y estos no permitirán que el usuario tenga sesiones abiertas al mismo tiempo en distintos dispositivos.

También cuenta con funciones de geo localización a través de GoogleMaps para ubicar las partidas de los equipos, poder llevar un control de la quema de calorías en los entrenamientos o en las partidas de los jugadores lo cual beneficiará a los jugadores en su trayectoria deportiva. Además, los usuarios pueden hacer un CRUD (Crear, Leer, Actualizar y Borrar) de los equipos, partidas, perfiles y torneos con ciertas restricciones dependiendo del caso, por otro lado se implementó un sistema de reputación por estrellas, lo cual fomentara la competitividad, también contará con un área para el control del desempeño del jugador en la cual podrá llevar un sistema de entrenamiento y quema de calorías con distintas opciones.

Además de lo mencionado cuenta con la mayoría de las características de las redes sociales que se manejan hoy en día como es el uso de recursos multimedia, vinculación de usuarios, almacenamiento de la información personal, entre otras. En cuanto a la metodología de desarrollo de software se utilizó la metodología XP y los lenguajes a utilizar para el desarrollo del software serán JAVA, PHP, CSS3, HTML5, MYSQL, JQUERY, JavaScript y como gestor de la base de datos se trabajará con MySQL.

CAPÍTULO II

MARCO TEÓRICO

Antecedentes de la Investigación

Como material de apoyo para el desarrollo de este proyecto se tomaron en cuenta trabajos previamente realizados por otros autores que contribuyan y/o complementen el proceso de alcanzar los objetivos y resultados esperados.

Facio P. (2017) en su proyecto de grado titulado “**NUTFIT, Herramienta Para Motivar Buenos Hábitos Alimenticios**” como requerimiento para la obtención del título de Ingeniero en Computación de la Universidad Nacional Autónoma de México en el cual se desarrolló de una aplicación Android para dispositivos móviles, cuya funcionalidad es motivar a los usuarios a adquirir buenos hábitos alimenticios, para esto se plantearon distintas situaciones en las que el usuario requiera realizar una medición de su Índice de Masa Corporal, la cantidad de calorías y gramos que puede perder al realizar una actividad física de su agrado.

También conocer la cantidad de algunas vitaminas que puede contener las frutas y las verduras que el cuerpo necesita día a día, opciones de alimentos saludables, recibir información de los beneficios de consumir vitaminas provenientes de frutas y las consecuencias de no consumirlas. El tipo de investigación para este proyecto es del tipo descriptiva y su objetivo general es “implementar una aplicación para teléfonos inteligentes mediante la cual los usuarios puedan mejorar su nutrición y quema de calorías”.

Dicha investigación concluye que la aplicación podría ayudar a reducir el número de pacientes que reciben las enfermedades de trastorno alimentario y sus consecuencias involucrar a las personas en transmitir la información proporcionada en esta aplicación a los demás miembros de su familia y conocidos para involucrar el hábito de consumir comida nutritiva, así se podrá tener más tiempo de vida. Se consultó esta información para aprender cómo

usar las fórmulas de quema de calorías y los factores que intervienen en la nutrición.

Asimismo, José M. y José A. (2016), en su trabajo especial de Grado titulado **“Desarrollo De Una Aplicación Móvil Sobre La Plataforma Android Para La Liga De Futbol Sala De La Facultad De Ciencias De La Universidad Central De Venezuela.”** para optar por el título de Licenciado en Computación en la Universidad Central de Venezuela desarrollaron una aplicación para la organización y administración de la liga de futsal de la Facultad de Ciencias en la Universidad Central de Venezuela para dispositivos móviles y smartphones sobre el sistema operativo Android, que permita a los organizadores mantener informados en tiempo real a la población interesada en el desarrollo de la competencia.

La investigación realizada está bajo el modelo de investigación de una adaptación de AgilUs, que es un método de 4 etapas de desarrollo con actividades bien definidas, que trabajan de forma iterativa para alcanzar el producto final de manera satisfactoria. Sin embargo, dicha metodología fue modificada a las necesidades reales de este trabajo, considerando la cantidad de participantes del mismo, el tipo de trabajo que se está realizando, entre otros. Su objetivo general es “desarrollar una aplicación web y móvil nativa para la organización y administración de la liga de futsal de la Facultad de Ciencias en la Universidad Central de Venezuela para dispositivos móviles y smartphones sobre el sistema operativo Android”.

La misma concluye que Debido al desarrollo de esta aplicación se alcanzó un aporte significativo a nivel tecnológico en el área de gestión de actividades deportivas de la Facultad de Ciencias de la Universidad Central de Venezuela, que no contaba con un sistema confiable, accesible y seguro que permitiera gestionar o administrar todo el proceso que conlleva realizar un torneo de futsal. Desde la creación de la competencia, ingreso de los equipos y divulgación de los resultados de la liga, los cuales serán replicados en una

aplicación para dispositivos móviles con el sistema operativo Android. Dicho trabajo es de gran ayuda ya que, plantea diseñar la interfaz de la aplicación de forma que cumpla con los lineamientos para un diseño eficiente.

Por otra parte, Pliego G. (2013), en su trabajo de fin de grado titulado **“Desarrollo De Una Aplicación Generadora Y Lectora De Códigos QR Seguros En Android”** para optar por el título en ingeniería informática en la Universidad Carlos III de Madrid, desarrolló una aplicación para dispositivos móviles Android que permita generar y leer códigos de barras bidimensionales QR cuya información esté cifrada de forma que sólo pueda acceder a ella aquella persona a quién va dirigida. También la aplicación permite leer los códigos que se encuentren en imágenes almacenadas en la memoria del dispositivo móvil.

Dicha investigación concluye que la aplicación en el estado actual no es un producto terminado o cien por ciento funcional. Sino un primer acercamiento o un prototipo como se mencionó en el título de este documento, pero que este prototipo logra integrar cada uno de los módulos sobre 2 de las 3 plataformas móviles más difundidas del mercado haciendo que sea altamente disponible y permite hacer un seguimiento objetivo del desarrollo de este proyecto. Lo cual ratifica la factibilidad técnica y el potencial funcional de una herramienta de esta naturaleza. Adicionalmente se revelaron las ventajas y falencias que no se contemplaron en el análisis, para así corregirlas en implementaciones posteriores.

Se subraya que, del análisis sistemático y buenos fundamentos de arquitectura empleados, se logró obtener una estructura de aplicación robusta y flexible que permite reusabilidad, extensibilidad y fácil mantenimiento en el código fuente. Se consultó esta investigación para aprender el funcionamiento de la lectura de códigos bidimensionales (código QR).

De igual forma, Oscar J. y Sonia E. (2015), presentaron una monografía titulada: **“Seguridad En Dispositivos Móviles Android”** para optar al título de Esp. En Seguridad Informática de la Universidad Nacional Abierta y a Distancia (UNAD). El objetivo principal fue Implementar políticas de seguridad de la información en dispositivos móviles con sistema operativo Android. Tal investigación tiene como fin analizar las debilidades que pueden presentarse en el sistema operativo Android para posteriormente presentar recomendaciones a los usuarios de estos dispositivos con el objetivo de prevenir ataques que puedan permitir el robo, manipulación o pérdida de la información que con estos dispositivos se pueda manejar.

Dicha investigación se fundamentó en un diseño no experimental y del tipo campo, bajo la modalidad de proyecto factible y se concluyó que todo sistema por más avanzado que esté presenta vulnerabilidades, sean pocas o muchas, y que como usuarios y especialistas en seguridad informática se deben tomar medidas preventivas y sensibilizar a las personas en nuestro entorno para ser conscientes sobre el uso seguro para este tipo de dispositivos. ESTA investigación es de utilidad, porque REFIERE los puntos relacionados sobre la seguridad del SO Android y como poder prevenir fallas en ella

Y por último se encuentra Javier B. (2014), de la Universidad Pablo de Olavide, quien realizó una tesis doctoral titulada: **“Análisis De Los Desplazamientos A Muy Alta Velocidad En Fútbol Profesional Mediante Tecnología GPS”**. Los aspectos más relevantes que tenemos para ilustrar el trabajo es la obtención de datos con GPS en partidos de competición no oficial de un equipo profesional de la Primera División de la Liga de Fútbol Profesional de España. Además, también se monitorizó toda una temporada de entrenamientos del mismo equipo, permitiendo establecer una comparación entre ambas situaciones (competición y entrenamientos).

La mencionada investigación concluye que con los datos que se aportan en el estudio sobre las deficiencias que hay entre partidos de competición y los

entrenamientos para desarrollar acciones de muy alta velocidad en los futbolistas, donde los espacios ocupan un papel destacado en las sesiones de entrenamientos, pero que no son las más eficientes para conseguir los estímulos necesarios que demanda un futbolista para desarrollar el número de veces que necesita un jugador ejercitarse en sprint de muy alta velocidad durante un partido de fútbol. Esta da un aporte significativo sobre la relación existente entre la distancia que recorren los jugadores y su entrenamiento.

Bases Teóricas

2.1.1 Framework.

En el concepto framework se emplea muchos ámbitos del desarrollo de sistemas software, no sólo en el ámbito de aplicaciones Web. Podemos encontrar framework para el desarrollo de aplicaciones médicas, de visión por computador, para el desarrollo de juegos, y para cualquier ámbito que pueda ocurrir. En general, el término framework, se refiere a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Un framework Web, por tanto, podemos definirlo como un conjunto de componentes (por ejemplo, clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web.

2.1.2 Aplicación Web

La Universidad de Alicante (2007) sostiene que: En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios

pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bien conocidos de aplicaciones web.

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

2.1.3 Estructura de las aplicaciones web

Aunque existen muchas variaciones posibles, una aplicación web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web ofrece la primera capa, y un motor capaz de usar alguna tecnología web dinámica, por ejemplo: PHP, Java Servlets o ASP, ASP.NET, CGI, ColdFusion, embPerl, Python o Ruby onRails que constituye la capa intermedia. Por último, una base de datos constituye la tercera y última capa.

El navegador web manda peticiones a la capa intermedia que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario. En la actualidad se ha vuelto común el uso de aplicaciones web para el manejo de información debido que hace más

fácil de modificar y corregir sin necesidad de afectar el resto de la funcionalidad de la página.

Ya que el sistema posee una parte de aplicación Web es necesario su conocimiento y sus funcionalidades para así poder aplicarlas al mismo.

2.1.4 API

La interfaz de programación de aplicaciones, abreviada como API del inglés: Application Programming Interface, es un conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Tiene como función establecer una comunicación con una base de datos, un sistema operativo o un protocolo de comunicaciones, por citar algunas posibilidades. Incluso las redes sociales emplean distintas API: una herramienta para gestionar tuits, por ejemplo, puede utilizar una API para establecer una comunicación con los servidores que utiliza Twitter.

Otra de las claves del funcionamiento de las API es la facilidad de integración. Estas herramientas tienen que resultar simples de integrar a otro software para que las comunicaciones puedan desarrollarse con éxito. De igual manera, sus actualizaciones no deberían generar conflictos para que su labor siga siendo óptima. Para evitar los problemas relacionados con sus actualizaciones, es necesario que se cumpla uno de los siguientes dos puntos, o bien ambos: que el formato de los datos que envía como respuesta al programa en el cual se integra no cambie; que la documentación explique con mucha claridad cómo adaptar el programa para que pueda continuar utilizando dichos datos.

2.1.5 Base de Datos

Según Silberschatz, A., Korth, H. y Sudarshan, S. (2002). Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo

contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Las bases de datos pueden clasificarse de varias maneras, de acuerdo al contexto que se esté manejando, la utilidad de las mismas o las necesidades que satisfagan. Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar de acuerdo a su modelo de administración de datos.

Un modelo de datos es básicamente una "descripción" de algo conocido como contenedor de datos (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de base de datos; por lo general se refieren a algoritmos, y conceptos matemáticos.

Es importante tener en cuenta el uso de una base de datos para el resguardo de la información ya que este proyecto contará con una amplia cantidad de información sobre los estudiantes y es necesario poder resguardarla para poder acceder a ella cuando sea necesario.

2.1.6 Android

David Orozco (2014) sostiene que Android es un sistema operativo basado en el kernel de Linux diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas, y también para relojes inteligentes, televisores y automóviles, inicialmente desarrollado por Android, Inc. Google respaldó económicamente y más tarde compró esta empresa en 2005. Android fue presentado en 2007 junto la fundación del Open Handset Alliance: un consorcio de compañías de hardware, software y telecomunicaciones para avanzar en los estándares abiertos de los dispositivos móviles.

Características y Especificaciones Actuales:

- **Diseño de dispositivo:** La plataforma es adaptable a pantallas de mayor resolución, VGA, biblioteca de gráficos 2D, biblioteca de gráficos 3D basada en las especificaciones de la OpenGL ES 2.0 y diseño de teléfonos tradicionales.
- **Almacenamiento:** SQLite, una base de datos liviana, que es usada para propósitos de almacenamiento de datos.
- **Conectividad:** Android soporta las siguientes tecnologías de conectividad: GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, HSDPA, HSPA+, NFC y WiMAX. GPRS, UMTS, HSPA+ Y HSDPA+
- **Mensajería:** SMS y MMS son formas de mensajería, incluyendo mensajería de texto y ahora la Android Cloud to Device Messaging Framework (C2DM) es parte del servicio de Push Messaging de Android.
- **Navegador web:** El navegador web incluido en Android está basado en el motor de renderizado de código abierto WebKit, emparejado con el motor JavaScript V8 de Google Chrome. El navegador por defecto de Ice Cream Sandwich obtiene una puntuación de 100/100 en el test Acid3.
- **Soporte de Java:** Aunque la mayoría de las aplicaciones están escritas en Java, no hay una máquina virtual Java en la plataforma. El bytecode Java no es ejecutado, sino que primero se compila en un ejecutable Dalvik y corre en la Máquina Virtual Dalvik. Dalvik es una máquina virtual especializada, diseñada específicamente para Android y optimizada para dispositivos móviles que funcionan con batería y que tienen memoria y procesador limitados. El soporte para J2ME puede ser agregado mediante aplicaciones de terceros como el J2ME MIDP Runner.

- **Soporte Multimedia:** Android soporta los siguientes formatos multimedia: WebM, H.263, H.264 (en 3GP o MP4), MPEG-4 SP, AMR, AMR-WB (en un contenedor 3GP), AAC, HE-AAC (en contenedores MP4 o 3GP), MP3, MIDI, OggVorbis, WAV, JPEG, PNG, GIF y BMP.
- **Soporte para Streaming:** Streaming RTP/RTSP (3GPP PSS, ISMA), descarga progresiva de HTML (HTML5 <video>tag). Adobe Flash Streaming (RTMP) es soportado mediante el Adobe Flash Player. Se planea el soporte de Microsoft SmoothStreaming con el port de Silverlight a Android. Adobe Flash HTTP DynamicStreaming estará disponible mediante una actualización de Adobe Flash Player.
- **Soporte para Hardware Adicional:** Android soporta cámaras de fotos, de vídeo, pantallas táctiles, GPS, acelerómetros, giroscopios, magnetómetros, sensores de proximidad y de presión, sensores de luz, gamepad, termómetro, aceleración por GPU 2D y 3D.
- **Entorno de desarrollo:** Incluye un emulador de dispositivos, herramientas para depuración de memoria y análisis del rendimiento del software. El entorno de desarrollo integrado es Eclipse (actualmente 3.4, 3.5 o 3.6) usando el plugin de Herramientas de Desarrollo de Android.
- **Google Play:** Google Play es un catálogo de aplicaciones gratuitas o de pago en el que pueden ser descargadas e instaladas en dispositivos Android sin la necesidad de un PC.
- **Multi-táctil:** Android tiene soporte nativo para pantallas capacitivas con soporte multi-táctil que inicialmente hicieron su aparición en dispositivos como el HTC Hero. La funcionalidad fue originalmente desactivada a nivel de kernel (posiblemente para evitar infringir patentes de otras compañías). Más tarde, Google publicó una actualización para el NexusOne y el Motorola Droid que activa el soporte multi-táctil de forma nativa.

- **Bluetooth:** El soporte para A2DP y AVRCP fue agregado en la versión 1.5; el envío de archivos (OPP) y la exploración del directorio telefónico fueron agregados en la versión 2.0; y el marcado por voz junto con el envío de contactos entre teléfonos lo fueron en la versión 2.2
- **Video-llamada:** Android soporta video-llamada a través de Google Talk desde su versión HoneyComb.
- **Multitarea:** Multitarea real de aplicaciones está disponible, es decir, las aplicaciones que no estén ejecutándose en primer plano reciben ciclos de reloj.
- **Características basadas en voz:** La búsqueda en Google a través de voz está disponible como "Entrada de Búsqueda" desde la versión inicial del sistema.
- **Tethering:** Android soporta tethering, que permite al teléfono ser usado como un punto de acceso alámbrico o inalámbrico (todos los teléfonos desde la versión 2.2, no oficial en teléfonos con versión 1.6 o inferiores mediante aplicaciones disponibles en Google Play (por ejemplo PdaNet). Para permitir a un PC usar la conexión de datos del móvil android se podría requerir la instalación de software adicional.

Arquitectura.

Los componentes principales del sistema operativo de son:

- **Aplicaciones:** las aplicaciones base incluyen un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y otros. Todas las aplicaciones están escritas en lenguaje de programación Java.
- **Marco de trabajo de aplicaciones:** los desarrolladores tienen acceso completo a los mismos APIs del framework usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes; cualquier aplicación puede publicar sus capacidades y

cualquier otra aplicación puede luego hacer uso de esas capacidades (sujeto a reglas de seguridad del framework). Este mismo mecanismo permite que los componentes sean reemplazados por el usuario.

- **Bibliotecas:** Android incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android; algunas son: System C library (implementación biblioteca C estándar), bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite, entre otras.
- **Runtime de Android:** Android incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrito de forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente. Dalvik ejecuta archivos en el formato DalvikExecutable (.dex), el cual está optimizado para memoria mínima. La Máquina Virtual está basada en registros y corre clases compiladas por el compilador de Java que han sido transformadas al formato.dex por la herramienta incluida "dx".
- **Núcleo Linux:** Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software. Este tipo de tecnología, tiene un gran impacto en la sociedad hoy en día permitiendo a los usuarios tener un fácil acceso a la información en la web y su conocimiento e implementación son necesarios para que los estudiantes tengan todas las comodidades para utilizar el sistema.

2.1.7 PHP

Según Mehdi Achour y otros (2009): PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se: podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo.

Fue creado originalmente por Rasmus Lerdorf en 1995. Actualmente el lenguaje sigue siendo desarrollado con nuevas funciones por el grupo PHP. Es importante tener este conocimiento, ya que es uno de los lenguajes más importantes para el desarrollo de páginas web y vinculación de base de datos y su conocimiento será necesario para la realización de este proyecto.

2.2.2 MYSQL

Tal como señala Nogueron Toledo Goretty (2011), es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca.

MySQL es un software de código abierto, licenciado bajo la GPL de la GNU, aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL.

El lenguaje de programación que utiliza MySQL es “Structured Query Language” (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales.

Características principales

Inicialmente, MySQL carecía de algunos elementos esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de esto, atrajo a los desarrolladores de páginas web con contenido dinámico, debido a su simplicidad, de tal manera que los elementos faltantes fueron complementados por la vía de las aplicaciones que la utilizan. Poco a poco estos elementos faltantes, están siendo incorporados tanto por desarrolladores internos, como por desarrolladores de software libre.

2.1.8 JAVA

Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes.

2.1.9 Metodología XP

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define

como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Características Fundamentales

- **Desarrollo iterativo e incremental:** pequeñas mejoras, unas tras otras.
- **Pruebas unitarias continuas,** frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión. Se aconseja escribir el código de la prueba antes de la codificación. Véase, por ejemplo, las herramientas de prueba JUnit orientada a Java, DUnit orientada a Delphi, NUnit para la plataforma.NET o PHPUnit para PHP. Estas tres últimas inspiradas en JUnit, la cual, a su vez, se inspiró en SUnit, el primer framework orientado a realizar tests, realizado para el lenguaje de programación Smalltalk.
- **Programación en parejas:** se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera -el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata.
- Frecuente **integración del equipo de programación con el cliente** o usuario. Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.
- **Corrección de todos los errores** antes de añadir nueva funcionalidad. Hacer entregas frecuentes.
- **Refactorización del código,** es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento. Las pruebas han de garantizar que en la refactorización no se ha introducido ningún fallo.
- **Propiedad del código compartida:** en vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo

distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.

- **Simplicidad** en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo.

Actividades de Xp

- **Codificar**

Es necesario codificar y plasmar nuestras ideas a través del código. En programación, el código expresa la interpretación del problema, así podemos utilizar el código para comunicar, para hacer comunes las ideas, y por tanto para aprender y mejorar.

- **Hacer pruebas**

Las características del software que no pueden ser demostradas mediante pruebas simplemente no existen. Las pruebas dan la oportunidad de saber si lo implementado es lo que en realidad se tenía en mente. Las pruebas nos indican que nuestro trabajo funciona, cuando no podemos pensar en ninguna prueba que pudiese originar un fallo en nuestro sistema, entonces habremos acabado por completo.

- **Escuchar**

En una frase, "Los programadores no lo conocemos todo, y sobre todo muchas cosas que las personas de negocios piensan que son interesantes. Si ellos pudieran programarse su propio software ¿para qué nos querrían?".

Si vamos a hacer pruebas tenemos que preguntar si lo obtenido es lo deseado, y tenemos que preguntar a quien necesita la información. Tenemos que escuchar a nuestros clientes cuáles son los problemas de su negocio, debemos de tener una escucha activa explicando lo que es fácil y difícil de obtener, y la realimentación entre ambos nos ayudan a todos a entender los problemas.

- **Diseñar**

El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben de ser sencillos, si alguna parte del sistema es de desarrollo complejo, lo apropiado es dividirla en varias. Si hay fallos en el diseño o malos diseños, estos deben de ser corregidos cuanto antes.

Prácticas Básicas de Xp

De forma aislada, cualquier práctica individual de Xp tiene poco sentido, pero en conjunto, unas compensan las carencias que las otras puedan tener.

- **El juego de la Planificación - (Planning Game)**

El alcance de la siguiente versión esta definido por las consideraciones de negocios (prioridad de los módulos, fechas de entrega) y estimaciones técnicas (estimaciones de funciones, consecuencias).

El objetivo del juego es maximizar el valor del software producido, La estrategia es poner en producción las características más importantes lo antes posible, Las Piezas clave son las Story Cards, Los Jugadores son los desarrolladores y el cliente y las Movidas son Exploración, Selección y Actualización.

- **Versiones Pequeñas (Short Releases)**

Un sistema simple se pone rápidamente en producción. Periódicamente, se producen nuevas versiones agregando en cada iteración aquellas funciones consideradas valiosas para el cliente

- **Metáfora del Sistema (Metaphor)**

Cada Proyecto es guiado por una historia simple de cómo funciona el sistema en general, reemplaza a la arquitectura y debe estar en lenguaje común, entendible para todos (Cliente y Desarrolladores), esta puede cambiar permanentemente.

- **Diseño Simple (Simple Designs)**

El sistema se diseña con la máxima simplicidad posible (YAGNY - "No vas a necesitarlo"), Se plasma el diseño en tarjetas CRC (Clase – Responsabilidad- Colaboración), no se implementan características que no son necesarias, con esta técnica, las clases descubiertas durante el análisis pueden ser filtradas para determinar qué clases son realmente necesarias para el sistema.

- **Pruebas Continuas (Testing)**

Los casos de prueba se escriben antes que el código. Los desarrolladores escriben pruebas unitarias y los clientes especifican pruebas funcionales.

- **Refactorización (Refactoring)**

Es posible reestructurar el sistema sin cambiar su comportamiento, por ejemplo eliminando código duplicado, simplificando funciones, Mejorando el código constantemente, si el código se esta volviendo complicado se debería modificar el diseño y volver a uno más simple. Refactoring (Modificar la forma del código sin cambiar su funcionamiento).

- **Programación por parejas (Pair Programming)**

El código es escrito por dos personas trabajando en el mismo computador. "Una sola maquina con un teclado y un mouse"

- **Posesión Colectiva del Código (Collective Code Ownership)**

Nadie es dueño de un modulo. Cualquier programador puede cambiar cualquier parte del sistema en cualquier momento, siempre se utilizan estándares y se excluyen los comentarios, Los test siempre deben funcionar al 100% para realizar integraciones con todo el código permanentemente.

- **Integración Continua (Continuous Integration)**

Los cambios se integran en el código base varias veces por día. Todos los casos de prueba se deben pasar antes y después de la integración, se dispone de una máquina para la integración y se realizan test funcionales en donde participa el cliente.

- **Semana Laboral de 40 Horas (40-Hour Week)**

Cada Trabajador trabaja no más de 40 Horas por semana. Si fuera necesario hacer horas extra, esto no debería hacerse dos semanas consecutivas. Sin héroes, esto hace que se reduzca la rotación del personal y mejora la calidad del producto.

- **Cliente en el Sitio (On Site Customer)**

El equipo de desarrollo tiene acceso todo el tiempo al cliente, el cual esta disponible para responder preguntas, fijar prioridades, etc. Esto no siempre se consigue; Un cliente muy Junior no sirve y un cliente muy Sénior no es disponible. "Lo ideal es un cliente Analista".

- **Estándares de Codificación (Coding Standard)**

Todo el código debe estar escrito de acuerdo a un estándar de codificación

Definición de Términos Básicos

- **Interfaz de usuario:** La interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina, equipo, computadora o

dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo.

- **Página Web:** Una página web, página electrónica o ciber página, es un documento o información electrónica capaz de contener texto, sonido, vídeo, programas, enlaces, imágenes, y muchas otras cosas, adaptada para la llamada World Wide Web (WWW) y que puede ser accedida mediante un navegador.
- **Servidor Web:** Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o Aplicación del lado del cliente.
- **Software:** Es un conjunto de instrucciones o programas que le permiten al computador realizar tareas, este es de gran importancia ya que representa la interacción entre el usuario y el computador.
- **CRUD:** es el acrónimo de "Crear, Leer, Actualizar y Borrar" (del original en inglés: Create, Read, Update and Delete), que se usa para referirse a las funciones básicas en bases de datos o la capa de persistencia en un software.
- **Web-Service:** es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet.
- **Librería:** es un conjunto de implementaciones funcionales, codificadas en un lenguaje de programación, que ofrece una interfaz bien definida para la funcionalidad que se invoca.
- **Deploy:** realizar una instalación de un sistema o aplicación en todos los equipos que lo precisen, tanto en local como en remoto.

- **Springs:** se refiere a la ejecución de una rutina de velocidad, trabajando en intervalos en los cuales debes aumentar la velocidad y disminuir la distancia recorrida.
- **El índice de masa corporal (IMC):** es una razón matemática que asocia la masa y la talla de un individuo, ideada por el estadístico.
- **La Tasa de metabolismo basal (TMB):** es el valor mínimo de energía necesaria para que la célula subsista. Esta energía mínima es utilizada por la célula en las reacciones químicas intracelulares necesarias para la realización de funciones metabólicas esenciales, como es el caso de la respiración.
- **Vue.JS:** es un framework progresivo para construir interfaces de usuario. A diferencia de otros frameworks monolíticos, Vue está diseñado desde el inicio para ser adoptado incrementalmente.

CAPÍTULO III

MARCO METODOLÓGICO

Tipo de investigación

El siguiente trabajo de investigación se encuentra catalogado dentro del tipo de proyecto especial, la Universidad Pedagógica Experimental Libertador (UPEL) define proyecto especial como “el desarrollo de software, prototipos y productos tecnológicos en general”. Estos proyectos especiales se componen de actividades, uso de recursos materiales o financieros; requieren objetivos específicos, necesidades o problemas y se deben realizar en un tiempo determinado. Según El manual de Tesis de Grado y Especialización y Maestría y Tesis Doctorales de la Universidad Pedagógica Libertador (UPEL), (2006):

Trabajo que lleve a creaciones tangibles, susceptibles de ser utilizadas como soluciones a problemas demostrados, o que respondan a necesidades e intereses de tipo cultural. Se incluyen en esta categoría los trabajos de elaboración de libros de texto y de materiales de apoyo educativo, el desarrollo de software, prototipos y de productos tecnológicos en general. (p. 22).

Diseño de la Investigación

Según los autores Santa P y Feliberto M (2010), se define que “El diseño no experimental es el que se realiza sin manipular en forma deliberada ninguna variable. El investigador no sustituye intencionalmente las variables independientes. Se observan los hechos tal y como se presentan en su contexto real y en un tiempo determinado o no, para luego analizarlos. Por lo tanto en este diseño no se construye una situación específica si no que se observa las que existen.” (pag.87).

Tomando lo antes mencionado, la presente investigación cuenta con un diseño no experimental, debido a que no se pretende modificar los indicadores recolectados por los deportistas, sino más bien estudiar el comportamiento de estos, de igual forma, solo se utilizaran fórmulas y tecnologías ya existentes.

Nivel de la investigación

De acuerdo con Arias F. (1999), se define que la investigación descriptiva como: “Los estudios descriptivos miden de forma independiente las variables y aun cuando no se formulen hipótesis, tales variables aparecen enunciadas en los objetivos de investigación.”. Por lo tanto esta investigación es del tipo descriptiva, ya que en esta se estudiara los factores que intervienen en el sedentarismo y dar respuesta a estos para mejorar la calidad de vida de los deportistas.

Población y Muestra

Para Arias. (2006) la población es “un conjunto finito o infinito de elementos con características comunes, para los cuales serán extensivas las conclusiones de la investigación. Esta queda limitada por el problema y por los objetivos del estudio.” (p.81), por otra parte Según Tamayo, T. Y Tamayo, M (1997), se afirma que la muestra “es el grupo de individuos que se toma de la población, para estudiar un fenómeno estadístico” (p.38).

Haciendo énfasis en lo antes mencionado, la población para esta investigación está conformada por los deportista del futbol del estado Carabobo tomando como muestra, escogidas al azar, cinco (5), canchas o complejos deportivos de futbol y dos (2) equipos rivales dentro de cada una de las mismas. Los cuales se encargaran de probar el sistema de quema de calorías y las tecnologías aplicadas, para así verificar que sea de utilidad para sus entrenamientos.

Técnicas e Instrumentos de recolección de datos

Arias (1999), menciona que “las técnicas de recolección de datos son las distintas formas de obtener información.” (p.53), por otra parte, según Tamayo y Tamayo (2008), la encuesta “es aquella que permite dar respuestas a problemas en términos descriptivos como de relación de variables, tras la recogida sistemática de información según un diseño previamente establecido

que asegure el rigor de la información obtenida”. (p.24), es por ello que se utilizara la encuesta de tipo Dicotómica para que los deportistas puedan evaluar la aplicación.

Fases metodológicas

- **FASE I: Diagnosticar situación actual del desempeño los deportistas en Valencia estado Carabobo mediante técnicas de observación.**

En esta fase se realizó el diagnóstico de la situación actual de como es el entrenamiento de los deportistas, tiempo de ejercicio, medios que usan para organizarse, lugares deportivos que frecuentan, tiempo que tienen como deportistas y subdivisión por edades en las que se encuentran. Para lograr el desarrollo de esta fase se utilizó la observación directa y se aplicaron las encuestas correspondientes.

- **FASE II: Determinar los requerimientos funcionales y no funcionales de la aplicación para el establecimiento de la funcionalidad de la aplicación, aplicando técnicas de recolección de datos.**

En esta fase se realizó el diseño de la aplicación, usando la metodóloga XP, se modelaron los requerimientos mediante esquemas UML basándose en las funcionalidades propuestas.

- **FASE III: Diseñar una aplicación multiplataforma mediante esquemas de modelos para las distintas partes de la aplicación.**

Mediante archivos XML dentro del editor Android Studio móvil y por medio de algún Framework de diseño de páginas web se maquetó la aplicación multiplataforma.

- **FASE IV: Desarrollar la aplicación multiplataforma a través de las herramientas de programación seleccionadas.**

En esta fase se construyó la aplicación multiplataforma mediante el editor Android Studio y algún Framework de PHP.

RESULTADOS

Fase I: Diagnosticar situación actual del desempeño los deportistas en Valencia estado Carabobo mediante técnicas de observación.

Se procedió a buscar información que aportara relevancia en el desempeño de los deportistas y por ende se logró establecer ciertas variables: el tipo de deporte, la intensidad del ejercicio, el peso, el sexo, la altura, el tiempo de ejercicio, el índice de masa corporal (IMC) y la tasa de metabolismo basal diario (TMB), adicionalmente se utilizaron técnicas de observación en los deportistas que arrojaron resultados determinantes para el diagnóstico de la problemática en cuestión.

Otra de las tareas a realizar fue, una entrevista en Gol Vinotinto ca, empresa encargada de alquilar canchas de futbol de grama sintética y de manejar una academia para el entrenamiento de este deporte. El dueño nos comentó que el sistema que utiliza para llevar el control de entrenamiento de su academia es manual y que sus jugadores tienen que estar en contacto con esta empresa para ver sus resultados. Además, señalaron los jugadores de esta entidad que no existía alguna plataforma dedicada exclusivamente a prestar un servicio gratuito para que los jugadores organicen sus encuentros.

4.1.1 Actividad I: Búsqueda de información

Se encontraron las fórmulas para cuantificar el desempeño de los deportistas y poder compararlas en periodos de tiempo. Se determinó que en primera instancia es importante hacer un diagnóstico inicial del **TMB** y **IMC** usando los datos de la persona, para el TMB **La fórmula de Harris-Benedict** revisadas por Mifflin y St Jeor (ver Formula 1) en combinación con su tabla de intensidad de ejercicio (ver Tabla 1), permitiendo determinar el consumo de las calorías necesarias por día para cada persona.

Fórmula 1:

Hombres	$TMB = (10 \times \text{peso en kg}) + (6,25 \times \text{altura en cm}) - (5 \times \text{edad en años}) + 5$
Mujeres	$TMB = (10 \times \text{peso en kg}) + (6,25 \times \text{altura en cm}) - (5 \times \text{edad en años}) - 161$

Fuente: ecuaciones de Harris-Benedict revisadas por Mifflin y St Jeor (1990)

Tabla 1:

Poco o ningún ejercicio	Calorías diarias necesarias = TMB x 1,2
Ejercicio ligero (1-3 días a la semana)	Calorías diarias necesarias = TMB x 1,375
Ejercicio moderado (3-5 días a la semana)	Calorías diarias necesarias = TMB x 1,55
Ejercicio fuerte (6-7 días a la semana)	Calorías diarias necesarias = TMB x 1,725
Ejercicio muy fuerte (dos veces al día, entrenamientos muy duros)	Calorías diarias necesarias = TMB x 1,9

Fuente: ecuaciones de Harris-Benedict

Por otro lado la organización mundial de la salud (OMS) establece una fórmula que permite diagnosticar de manera general si una persona se encuentra en el peso correspondiente a su altura, llamada índice de masa corporal o IMC (ver Formula 2) conjuntamente con una tabla en la cual se encuentran los rangos adecuados para el IMC (ver Tabla 2).

Fórmula 2:

$IMC = \frac{\text{Peso (kg)}}{\text{Estatura}^2 \text{ (Mts.)}}$
<p><i>Fuente: OMS. (2012)</i></p>

Tabla 2:

IMC	Situación
Menor 18,5	Bajo peso
18,5-24,9	Normopeso
25-26,9	Sobrepeso grado I
27-29,9	Sobrepeso grado II
30-34,9	Obesidad de tipo I
35-39,9	Obesidad de tipo II
40-49,9	Obesidad de tipo III (mórbida)
Mayor 50	Obesidad de tipo IV (extrema)

Fuente: ecuaciones de Harris-Benedict

Según el deporte que se practique y el tiempo de la actividad, demanda un nivel de desempeño mayor o menor, en consecuencia la quema de calorías dependerá directamente del tipo de actividad física y el tiempo del ejercicio. Por lo tanto según **Compendio de Actividades Físicas** se encontró la fórmula de quema de calorías (ver Formula 3) y dentro de esta el término **MET** (ver anexo B), que se usa para reflejar los costos de energía de cada tipo de ejercicio en particular.

Formula 3

$\text{MET} = \text{kcal} / \text{kg} / \text{hora}$
$\text{Kcal} = \text{MET} \times \text{kg} \times \text{hora}$
<i>Fuente: Compendium of Physical Activities (2011)</i>

Según los archivos latinoamericanos de nutrición órgano oficial de la sociedad latinoamericana de nutrición publicados en 2013, Se determinó el consumo de la población venezolana realizadas en las Encuestas de Seguimiento al Consumo de Alimentos (ESCA) 2010- 2012. Las mismas arrojaron que los valores de referencia de proteínas recomendados como ingesta segura en g/kg/día son:

- para niñas y niños, menores de 1 año 1,14 a 1,80.
- de 1 a 3 años 0,90 a 1,14.
- de 4 a 6 años 0,86 a 0,89.
- de 7 a 10 años 0,91 a 0,92.
- En adolescentes de 12 a 19 años, el promedio es de 0,88 y 1,07 para el sexo masculino y femenino respectivamente.
- En adultos de 20 a 59 años, se recomienda 0,83.
- en adultos mayores de 60 años 1,00 para hombres y mujeres.

La cantidad de proteínas necesarias en la dieta se suele medir en gramos de proteínas a ingerir por kilogramos de peso corporal en un día. Como vemos, no se supera la barrera de los 2 g/kg/día (ver Tabla 3), donde muchas investigaciones concluyen que el cuerpo no es capaz de asimilar en un día más de esa cantidad. Aunque va a influir mucho el reparto que hagamos de esas proteínas a lo largo del día, la capacidad muscular de cada sujeto de incorporar los aminoácidos a su estructura muscular y si se utilizan suplementos.

Tabla 3:

Población	Proteína (g/kg/día)
Deportista Adolescente o pesista	1,5-2
Deporte de fuerza	1,5-1,7
Deporte de resistencia	1,2-1,6
Deporte Recreativos	1-1,2
Sedentarismo	0,8-1

Fuente: Sociedad Latinoamericana de Nutrición (2013)

4.1.2 Actividad II: Entrevista con los deportistas para establecer sus prioridades en el entrenamiento

En la cancha de Gol Vinotinto ca, se habló con los jugadores durante un evento en el cual participaron aproximadamente diez (10) equipos de futbol. Se les preguntó a los jugadores como manejan sus resultados cuando juegan contra otros

equipos y también como controlan su rendimiento en la cancha, de manera general, respondieron que no tenían un control de la cantidad de partidas que jugaban ni tampoco de los resultados de esa partida, también que a la hora de realizar los entrenamientos ellos cuantificaban los resultados sin llevar ningún control, sino simplemente viendo los resultados del mismo.

Otro de las personas a las cual se les entrevistó fue con Edson Tortolero, jugador profesional de ligas internacionales y actualmente central del Carabobo futbol club, a quien se le planteó la idea de crear una aplicación en la que los jugadores puedan organizar sus partidas y tener un control de sus enteramientos. Después plantearle la idea, nos comentó que le parecía una idea excelente y que jamás escuchó algo similar, también que en su experiencia no conoce una aplicación dentro del futbol para que los jugadores profesionales y amateur puedan darse a conocer en este ámbito.

Fase II: Determinar los requerimientos funcionales y no funcionales de la aplicación para el establecimiento de la funcionalidad de la aplicación, aplicando técnicas de recolección de datos.

Luego de haber realizado la investigación respectiva para poder entrar en contexto, se necesita establecer las funcionalidades dentro de la aplicación, por lo que fue necesario en primera instancia establecer unos requerimientos para la aplicación, después consultar la opinión del posible público al que se quiere llegar, y por último establecer los requerimientos funcionales y no funcionales de manera detallada. Lo cual fue muy importante, ya que para poder construir la aplicación fue necesario tener una idea de las características que debe contener la aplicación.

4.1.3 Actividad I: Elaboración de los requerimientos funcionales y no funcionales de la aplicación.

En este punto de la investigación se tiene una idea de cómo se debería comportar la aplicación y por ende se elaboró el documento de definición de requerimientos (ver Anexo C-1, documentos de requerimientos). Se determinó que el

control para los entrenamientos de los jugadores es el área de aplicación en la cual se orientó el desarrollo del proyecto, por lo que ese fue el enfoque técnico que se le dio al mismo y seguidamente se procedió a realizar una encuesta que permitió la comprensión de las necesidades de los jugadores.

Además de esto, se procedió a revisar la documentación acerca de las tecnologías modernas disponibles para el entrenamiento de los deportistas, en donde se encontró información referente al área de estudio, en lo que concierne a lo más destacados como opciones principales la librería de android Zxing, la API de GoogleMaps, la teoría de nodos, la conversión de imágenes y los objetos predefinidas de cronometro.

4.1.4 Actividad II: aplicar técnicas de recolección de datos.

Para poder hacer un diagnóstico más preciso basado en la población que se desea llegar, se aplicó encuestas dicotómicas las cuales cuentan con ocho (8) preguntas (ver ANEXO A), teniendo como fin: la recolección de datos para establecer si las características planteadas dentro de la aplicación son necesarias en el desarrollo del programa, y así determinar los requerimientos para el control de entrenamiento que los jugadores puedan tener.

Preguntas:

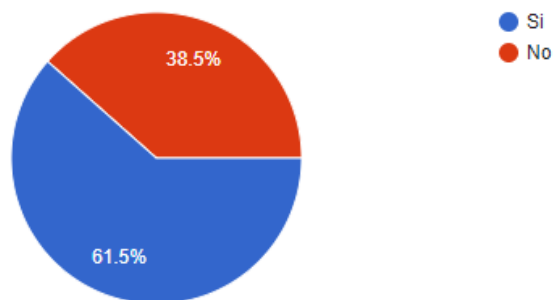
- Usted practica frecuentemente deporte.
- Considera que su salud es óptima.
- Cree usted que está en su peso ideal.
- Usa alguna aplicación para controlar los resultados de sus entrenamientos.
- Usaría una aplicación para organizar partidas de fútbol.
- Está familiarizado en el uso de alguna herramienta para la quema de calorías.
- Aspira ser un profesional del deporte.
- Usted usaria alguna aplicación para tener un control de sus entrenamientos.

4.1.5 Actividad III: análisis de los resultados de las encuestas

Se procedió a tabular las encuestas dicotómicas en graficas circulares las respuestas de los 39 participantes y analizar cada una de las mismas para aportar información más detallada. Esto con el fin de obtener los datos precisos para el análisis de las encuestas, ya que la aplicación necesita ser alimentada con los datos de una población real.

Usted practica frecuentemente deporte.

39 respuestas

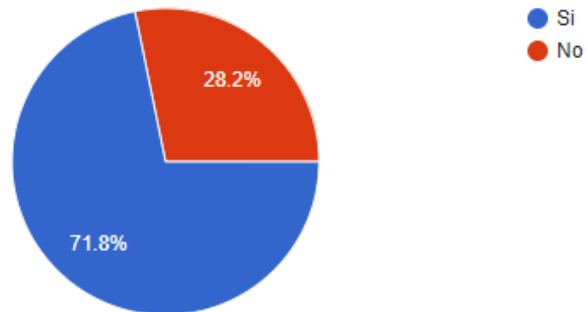


Fuente: Omar Perez (2018)

La respuesta a la sección “Usted practica frecuentemente deporte.”, Se totalizó con 61.5% de los jugadores que si, y un 38.5% los que no, esta pregunta se realizó con el fin de determinar si eran jugadores activos o no. Ya que en la aplicación se busca una población de jugadores activos que practiquen frecuentemente el deporte, para que puedan usar la misma como una herramienta más a la hora de practicar el futbol. Por ende se evidencia que hay una gran cantidad de jugadores activos como población, a los cuales la aplicación les puede ser de gran utilidad.

Considera que su salud es óptima.

39 respuestas

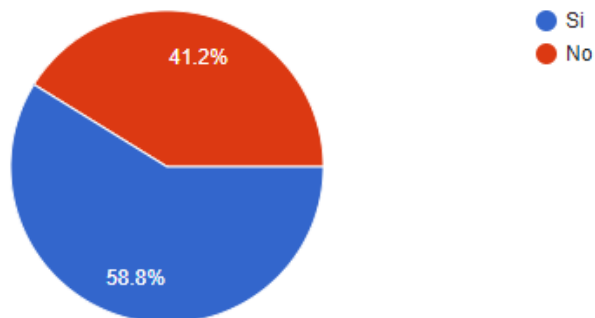


Fuente: Omar Perez (2018)

También a los encuestados se le consultó: “Considera que su salud es óptima.”, Se totalizó con 71.8% de los jugadores que si, y un 28.2% los que no, esta pregunta se realizó con el fin de evidenciar que el futbol es un deporte que mantiene a un gran porcentaje de los jugadores en una salud óptima. Por lo tanto escoger el futbol como deporte dentro la aplicación es una decisión correcta.

Cree usted que está en su peso ideal.

39 respuestas



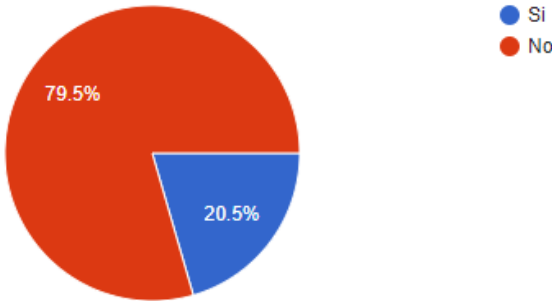
Fuente: Omar Perez (2018)

La respuesta a la pregunta: “Cree usted que está en su peso ideal.”, Se totalizó con 58.8% de los jugadores que si, y un 41.2% los que no, esta

pregunta se realizó para de establecer la percepción de los jugadores con respecto a su peso. Con estos resultados se determinó que la herramienta de control de calorías puede serles útiles a una gran parte de la población deportiva en el ámbito del futbol.

Usa alguna aplicación para controlar los resultados de sus entrenamientos.

39 respuestas

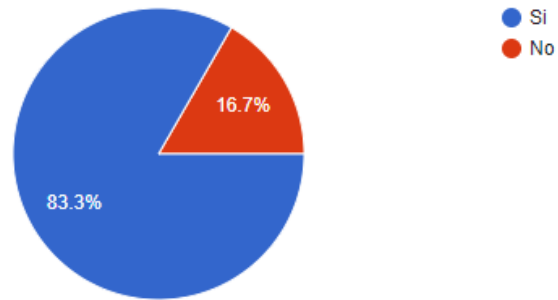


Fuente: Omar Perez (2018)

Se consultó a los encuestados con la pregunta: “Usa alguna aplicación para controlar los resultados de sus entrenamientos.”. Se totalizó con 20.5% de los jugadores que si, y un 79.5% los que no, esta pregunta se realizó con el fin de establecer si los jugadores utilizan alguna aplicación como apoyo de sus entrenamientos. Con estos resultados se determinó que la gran mayoría de los jugadores no utiliza y por ende es un obstáculo que se debe determinar la manera de como motivar a los jugadores a usarla.

Usaría una aplicación para organizar partidas de fútbol

39 respuestas

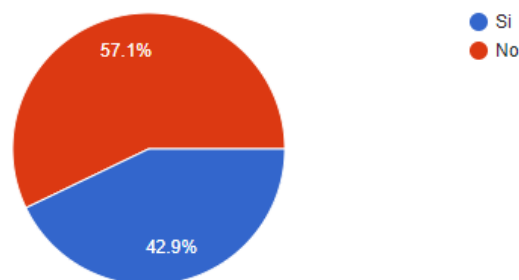


Fuente: Omar Perez (2018)

La respuesta a la interrogante “Usaría una aplicación para organizar partidas de fútbol.”. Se totalizó con 80.3% de los jugadores que si, y un 16.7% los que no, esta pregunta se realizó con el fin de determinar si los jugadores estarían dispuestos a usar la aplicación propuesta a desarrollar. Con estos resultados se estableció que una gran mayoría de los jugadores les motivaría usar la aplicación para organizar sus partidas de futbol.

Está familiarizado en el uso de alguna herramienta para la quema de calorías.

39 respuestas



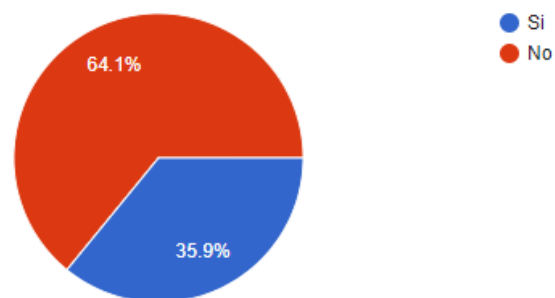
Fuente: Omar Perez (2018)

A los encuestados se le pregunto: “Está familiarizado en el uso de alguna herramienta para la quema de calorías.”. Se totalizó con 42.9% de los

jugadores que si, y un 57.1% los que no, esta pregunta se realizó con el fin de establecer si a los jugadores le seria difícil usar la herramienta de quema de calorías propuesta en la aplicación. Con estos resultados se estableció que se debe familiarizar a los jugadores a usar la herramienta de quema de calorías.

Aspira ser un profesional del deporte.

39 respuestas

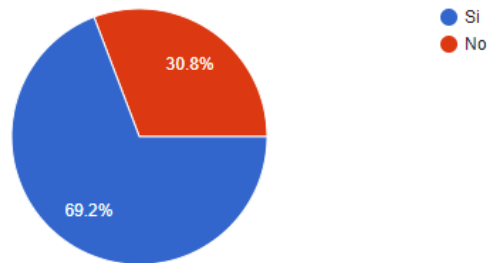


Fuente: Omar Perez (2018)

La respuesta a la pregunta “Aspira ser un profesional del deporte.”. Se totalizó con 35.9% de los jugadores que si, y un 64.1% los que no, esta pregunta se realizó con el fin de dar una cifra de la cantidad de jugadores que aspiran una carrera en el deporte. Se determinó que la aplicación puede apoyar a los jugadores a darse a conocer en este ámbito mediante la comunidad.

Usted usaria alguna aplicación para tener un control de sus entrenamientos.

39 respuestas



Fuente: Omar Perez (2018)

La respuesta a la pregunta “Usted usaria alguna aplicación para tener un control de sus entrenamientos.”. Se totalizó con 69.2% de los jugadores que si, y un 30.8% los que no, esta pregunta se realizó con el fin de establecer si los jugadores estarían dispuestos a incluir la tecnología de la aplicación en sus entrenamientos. Con estos resultados se evidenció que los jugadores son propensos al cambio siempre y cuando este sea beneficioso.

4.1.6 Actividad IV: Especificación de requerimientos funcionales y no funcionales.

Se comenzó con la redacción del documento de especificación de requerimientos, el cual está conformado por dos secciones. La primera consta de una tabla en donde se muestra la información del proyecto, tal como: el título, la descripción, la fecha de inicio y la versión del software a desarrollar. Luego, se muestra una segunda tabla en donde cada fila representa a un requerimiento del sistema, estos constituyen las características funcionales y no funcionales que deben implementarse en el software (ver Anexo C-2, documentos de requerimientos).

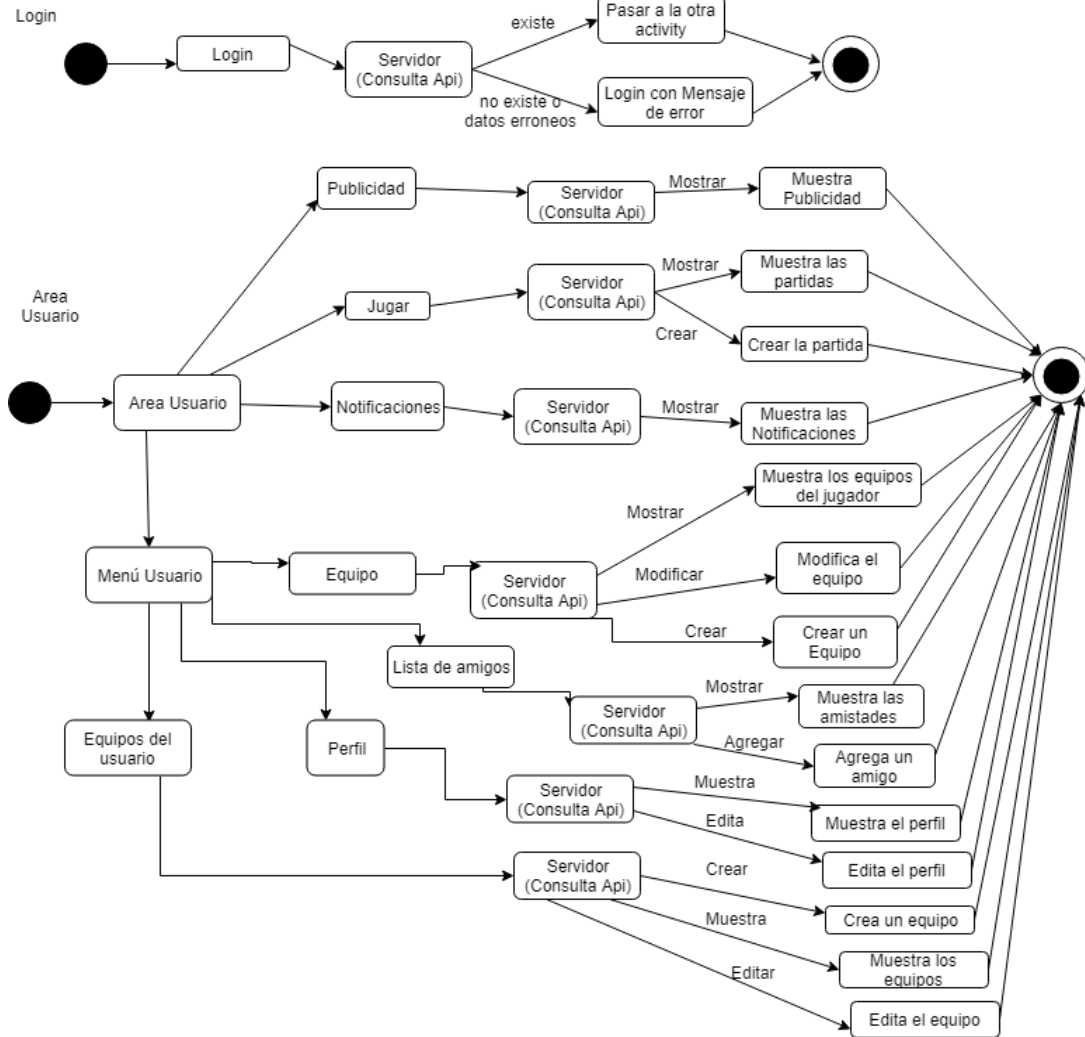
FASE III: Diseñar una aplicación multiplataforma mediante esquemas de modelos para las distintas partes de la aplicación.

En el desarrollo de aplicaciones, además de la funcionalidad, la utilidad y el correcto funcionamiento nos topamos con otras dos características imprescindibles: el diseño de la aplicación y la usabilidad. Para llegar a un diseño y usabilidad óptimos, debemos tener en cuenta el tipo de dispositivo, el sistema operativo y la imagen corporativa de la empresa. En el desarrollo de aplicaciones es imprescindible tener estos parámetros en mente antes de empezar a crear la aplicación. Por ello se escogió realizar bocetos minimalistas para que la aplicación Web y Android sean sencillas de usar para cualquier usuario. Además cabe destacar que no es necesarios realizar los bocetos Web ya que se verá muy similar con la aplicación Android.

4.1.7 Actividad I: Diseñar los distintos esquemas de modelos para las partes de la aplicación.

Según El Objeto Management Group (OMG), el cual supervisa la definición y el mantenimiento de las especificaciones de UML, menciona que el UML ofrece a los ingenieros y programadores la capacidad de usar un lenguaje para muchos propósitos durante todas las etapas del ciclo de vida del software en sistemas de cualquier tamaño. Por ende se procedió a realizar un diagrama de estado con el fin de presentar ideas claras en el desarrollo de la aplicación.

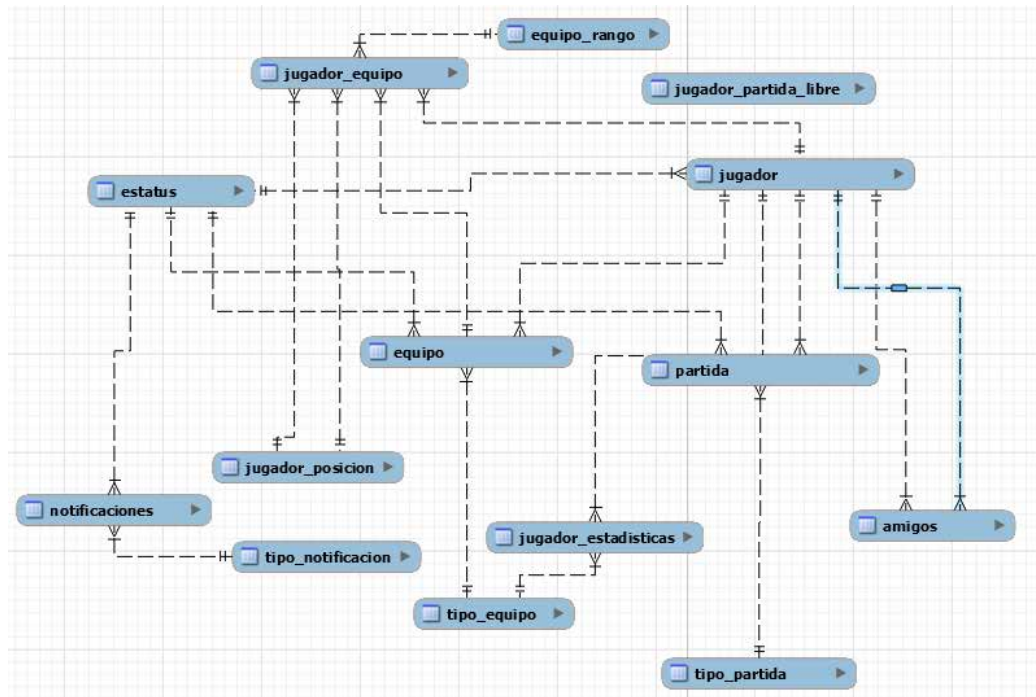
Diagrama 1:



Fuente: Omar Perez (2018)

El Modelo Entidad Relación (ER) permite desarrollar un diseño de base de datos en un esquema de alto nivel conceptual sin considerar los problemas de bajo nivel como la eficiencia, el modelo implícito del administrador de base de datos o las estructuras físicas de los datos”. Por esto se decidió realizar el diagrama E-R (ver diagrama 2), el cual permitió dar un punto de inicio para el desarrollo de la base de datos en MySQL.

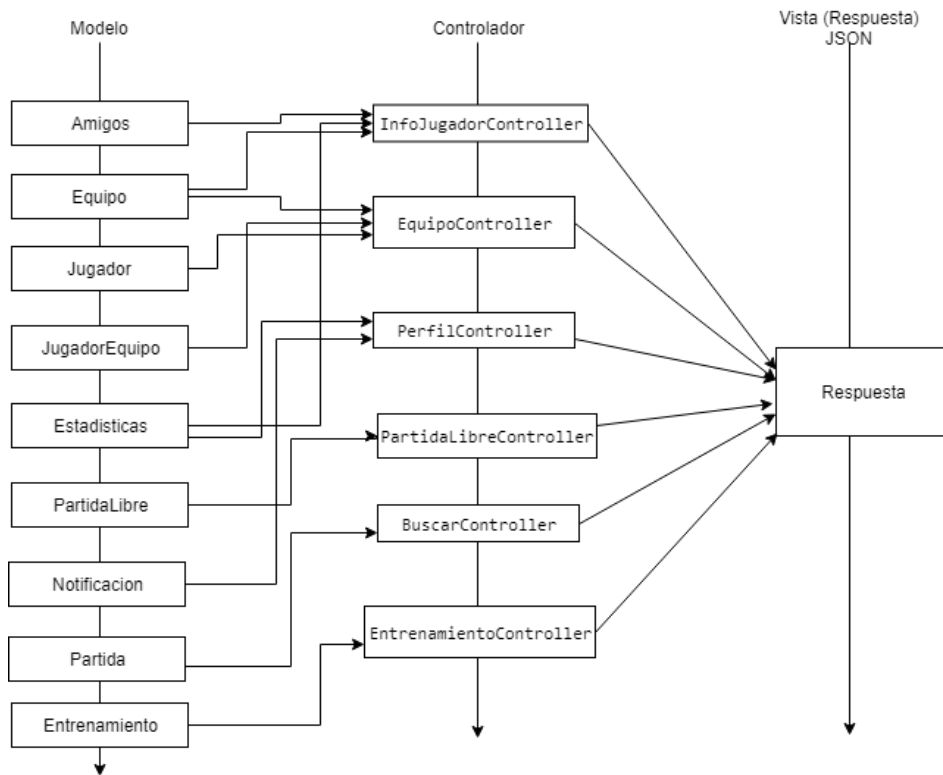
Diagrama 2:



Fuente: Omar Perez (2018)

El patrón de diseño de modelo-vista-controlador (MVC) especifica que una aplicación consta de un modelo de datos, de información de presentación y de información de control. El modelo contiene únicamente los datos puros de aplicación. La vista presenta al usuario los datos del modelo y sabe cómo acceder a los datos del modelo, pero no sabe el significado de estos datos ni lo que el usuario puede hacer para manipularlos y por último, el controlador está entre la vista y el modelo. Escucha los sucesos desencadenados por la vista (u otro origen externo) y ejecuta la reacción apropiada a estos sucesos. La mayoría de las aplicaciones hoy en día siguen este patrón por sus beneficios y sencillez una vez que se entienden los conceptos MVC. Por estas razones se decidió crear el diagrama de MVC para la API (ver diagrama 3).

Diagrama 3:



Fuente: Omar Perez (2018)

4.1.8 Actividad II: Diseñar los bocetos preliminares de la interfaz de la aplicación.

Una vez establecido los diagramas UML, se entiende la estructura de la aplicación y por ende en esta actividad, se pretende describir detalladamente las características referentes a al diseño como lo son: colores, temas, fuentes, animaciones, estilos, elementos dentro del diseño, para así en la **fase IV** empezar a desarrollar la aplicación con una idea clara de cómo se debe ver.

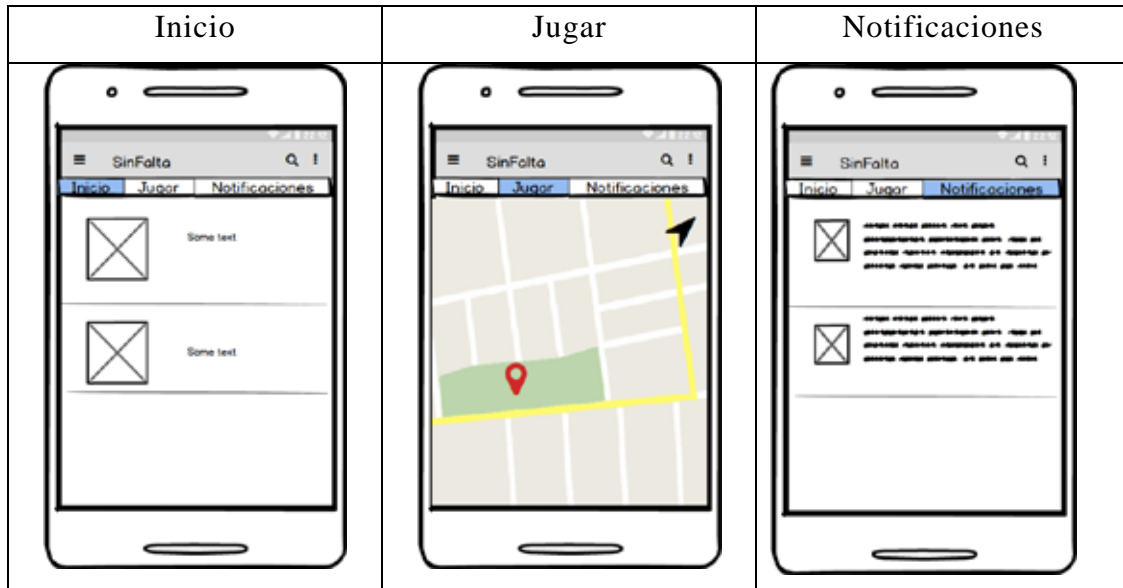
Área de Entrada



Fuente: Omar Perez (2018)

En el área entrada se procedió hacer un diseño en función al logo escogido, además cuenta con el mismo fondo, con imágenes referentes, botones y texto con un diseño elegante. Comenzando por el login: contiene el icono de la aplicación en la cabecera luego dos cajas de texto transparentes con animaciones de texto, dos botones uno de registro y otro para entrar a la aplicación en diferentes colores, y un texto para recuperar contraseña. Por otra parte el registro cuenta con cinco cajas de texto obligatorias y transparentes con animaciones de texto, un grupo de checkbox para seleccionar el sexo con un estilo elegante y el botón de registro en color. Y por último el recuperar contraseña con un botón y una caja de texto con animación.

Área de Usuario:



Fuente: Omar Perez (2018)

En el área de usuario cuenta con una barra de navegación con el nombre de la aplicación y 3 pestañas llamadas el Inicio, Jugar y Notificaciones. Comenzando por la pestaña de inicio está conformada por una lista en formatos de estilo cartas (estilo de diseño android) con un texto con el nombre de la publicidad y una descripción para cada una en colores llamativos. Por otra parte, la pestaña Jugar es completamente un mapa que cuenta con un estilo oscuro en azul para hacerlo más atractivo que posee un botón de ubicación en transparente. Y por último las notificaciones en forma de lista con animación desplazable con una imagen sobre el tipo de notificación en color gris, un texto con la descripción de la notificación y una fecha y hora en azul.

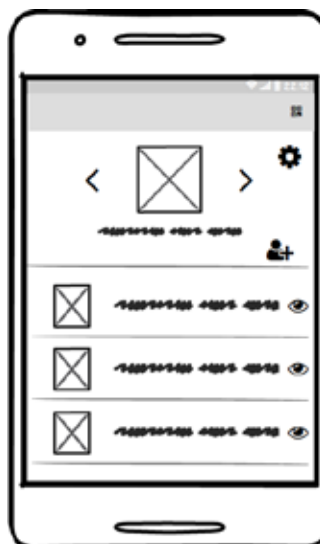
Perfil:



Fuente: Omar Perez (2018)

El perfil del jugador cuenta con la imagen del jugador en forma circular la cual se oculta al hacer deslizamiento en la pantalla, un icono pequeño en forma de código Q-R, unos botones con imágenes correspondientes al tipo de acción, una lista desplegable con diseño de carta y un texto de color verde para poder hacer las acciones necesarias.

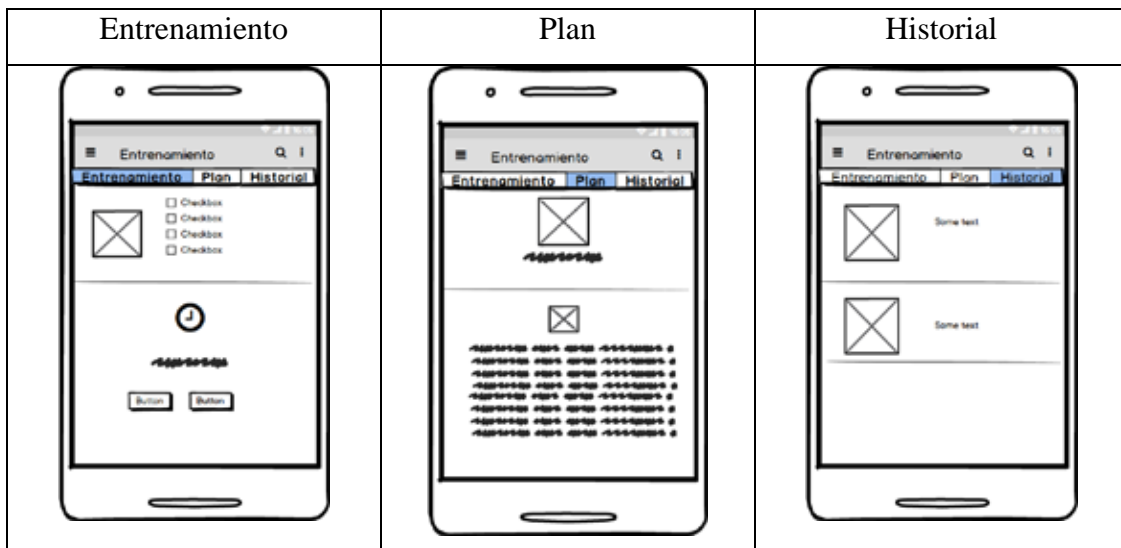
Mis equipos:



Fuente: Omar Perez (2018)

En la sección de mis equipos encontramos una imagen circular con la foto del equipo escogido, dos botones rojos para cambiar entre los equipos, otro botón con imagen de usuario en gris para agregar algún jugador al equipo ,un botón con forma de engranaje para realizar los ajustes, una lista completa con diseño basado en cartas para mostrar los jugadores de cada equipo, además de que cada jugador posee su foto en forma circular con un texto que posee el nombre del jugador y un botón en gris para poder ver su perfil.

Área Entrenamiento:



Fuente: Omar Perez (2018)

El área de entrenamiento encontramos una barra con opciones para navegar y un texto para saber en dónde se encuentra el usuario. Posee tres pestañas, entrenamiento, plan e historial. En las cuales el entrenamiento está diseñado con un cronometro personalizado el cual muestra el tiempo transcurrido en un texto y con dos botones iniciar/parar y reiniciar, además de contar con unas opciones de selección múltiple diseñadas atractivamente para el usuario. Por el lado del plan de entrenamiento un texto fácil de leer con una imagen referencial al texto y un botón con el tipo de entrenamiento que escogió la persona. Y por último el historial de los entrenamientos que cuenta con cartas que posee textos e imágenes referentes al entrenamiento llevado.

FASE IV: Desarrollar la aplicación multiplataforma a través de las herramientas de programación seleccionadas.

Para el desarrollo de la aplicación multiplataforma se presentan varias constantes, como lo son el lenguaje a elegir, los editores, los Frameworks y los hosting. En lo referente la aplicación constara por dos partes Frontend, la parte Web y la Android la cual están diseñada bajo los mismos bocetos de la fase III, y en el Backend una API del lado del servidor el cual se comunicara con la base de datos MySQL con su hosting en la plataforma de Heroku. Además cabe señalar que la aplicación Web ejecutara menos funciones pero guardando similitud a la parte Android

4.1.9 Actividad I: elegir las herramientas para el desarrollo de la aplicación.

Existen una inmensa variedad de herramientas de programación que se pueden elegir para satisfacer distintas necesidades dentro de la elaboración de la aplicación. Si bien es cierto que muchas de las herramientas se pueden utilizar en diferentes ámbitos, siempre suele haber algunas que destaque entre los demás para dicha área. Pero en este caso se eligieron por las ventajas que ofrecen y el manejo de las mismas.

Se eligió el editor de texto Android Studio ya que es el entorno de desarrollo integrado oficial para el desarrollo de aplicaciones. Además del potente editor de códigos y las herramientas para desarrolladores que ofrece, Android Studio ofrece aún más funciones que aumentan la productividad durante la compilación de apps.

Por el lado de la página Web se usó Vue.js el cual es un framework de JavaScript para construir interfaces de usuario. El cual está diseñado desde el inicio para ser adoptado incrementalmente. Por otro lado, Vue también es perfectamente capaz de soportar aplicaciones sofisticadas de una sola página (en inglés single-page-application o SPA) cuando se utiliza en combinación

con herramientas modernas y librerías compatibles. Además, de usar el editor de texto WebStorm de IntelliJ IDEA.

También en lo referente a la Api se implementó lumen el cual es micro-framework para PHP del tipo MVC el cual nos proporciona una interacción de datos de manera organizada y segura a la base de datos, y también a las partes frontend de la aplicación. Conjuntamente con el editor de texto PHPStorm de IntelliJ IDEA.

En lo referente a la base de datos para la aplicación multiplataforma se escogió MySQL por conocimientos prácticos en la misma y ser una de la más sencilla y con mejor documentación. Para el diseño y creación de la misma se usó la herramienta MySQL Workbench, la cual puede elaborar una representación visual de las tablas, vistas, procedimientos almacenados y claves foráneas de la base de datos. Además, es capaz de sincronizar el modelo en desarrollo con la base de datos real.

4.1.10 Actividad II: Implementar las nuevas tecnologías.

En la elaboración de esta aplicación se implementaron tecnologías de GPS, Código Q-R, Tokens de seguridad y Carga de imágenes. En lo relacionado con el GPS consiste en un sistema de localización el cual el usuario por medio de la aplicación muestra su ubicación actual y así poder encontrar las partidas para la práctica del futbol, por la parte del escaneo y visualización de códigos Q-R se usó la librería de ZXing bajo la licencia de apache la cual permite generar los códigos Q-R y la librería Barcode Scanner, también bajo la licencia de apache para leer los códigos Q-R.

En lo referente a los Tokens que se usaron para crear sesiones, se utilizó un una cadena de 64 dígitos generada por una función de PHP llamada base64_encode para encriptar las contraseñas de los usuarios y generar token de sesión, dándole así más fiabilidad y seguridad a la aplicación multiplataforma.

Para la carga de imágenes se implementó un algoritmo (ver algoritmo 1), el cual convierte una foto en formato bitmap para luego convertirla en una cadena de 64 bits (ver algoritmo2) y así subir las imágenes, ya que fue la forma que se consiguió para poder enviar las imágenes vía Http a la api y de la ahí al servidor Clouinary (hosting) el cual se encarga de reacomodar las imágenes en los distintos formatos conocidos y guardarlas.

Algoritmo 1:

```
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == CropImage.CROP_IMAGE_ACTIVITY_REQUEST_CODE) {
        CropImage.ActivityResult result = CropImage.getActivityResult(data);
        if (resultCode == RESULT_OK) {
            Uri resultUri = result.getUri();
            try {
                mBitmap = MediaStore.Images.Media.getBitmap(this.getContentResolver(), resultUri);
                final ProgressDialog progressDialog=new ProgressDialog( context: this);
                progressDialog.setMessage("Cargando");
                progressDialog.show();

                guardarFtoto(mBitmap,progressDialog);
            } catch (IOException e) {
                e.printStackTrace();
            }
            // Toast.makeText(this, resultUri.toString(), Toast.LENGTH_SHORT).show();
        } else if (resultCode == CropImage.CROP_IMAGE_ACTIVITY_RESULT_ERROR_CODE) {
            Exception error = result.getError();
            Log.d( tag: "ERROR===",error.toString());
        }
    }
}
```

Fuente: Omar Perez (2018)

Algoritmo 2:

```
public String getStringImage(Bitmap bmp){
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    bmp.compress(Bitmap.CompressFormat.PNG, quality: 100, baos);
    byte[] imageBytes = baos.toByteArray();
    return Base64.encodeToString(imageBytes, Base64.DEFAULT);
}
```

Fuente: Omar Perez (2018)

4.1.11 Actividad III: Desarrollo de la aplicación multiplataforma

Para el desarrollo de la aplicación se implementó la librería Volley en todas las ventanas que se comunican con el servidor, debido a que la librería Volley es una librería desarrollada por Google para optimizar el envío de peticiones Http desde las aplicaciones Android hacia servidores externos. Este componente actúa como una interfaz de alto nivel, liberando al programador de la administración de hilos y procesos tediosos de parsing, para permitir publicar fácilmente resultados en el hilo principal. Por ende cada vez que se mencione situaciones referentes a la comunicación de datos al servidor estaremos usándola. Además cabe destacar que el flujo de la información es recibido y enviado en formato JSON.



En esta ventana se implementó una validación que nos proporciona AndroidStudio en los cuadros de texto correspondiente al tipo de dato, como lo es el correo que debe llenar un formato de secuencia de caracteres y la clave que un dato no visible. Por otro lado encontramos las transiciones que son los

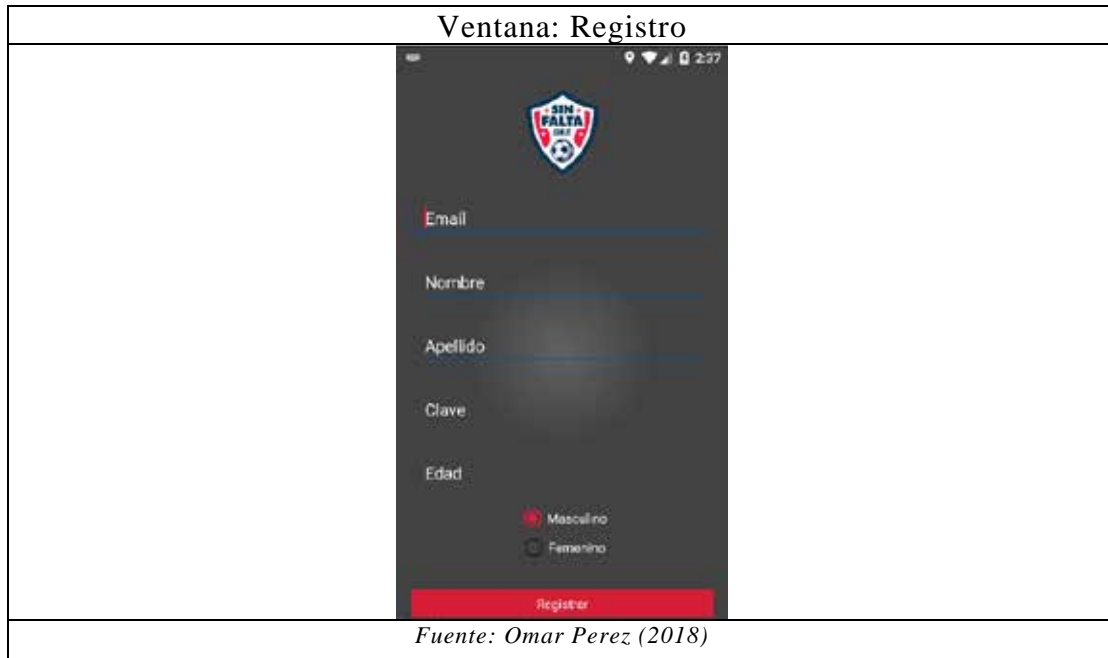
saltos de vista, esta ventana posee dos botones con oyentes de evento los cuales al hacer clic nos conduce por una parte el registro al área de registro y el olvido contraseña a la recuperación de la misma, esto es posible gracias a la clase Intent() predefinida por java. Por ultimo encontramos que al hacer clic en el login el cual tiene su oyente nos valida los campos y envía la información al servidor por el método POST. Cuando llega la petición al API verifica mediante un algoritmo (ver algoritmo 1) que no sean nulos los campos y si existe el jugador, para después comunicarse a la BD y encontrar el token o sesión del usuario, para así devolverlo a la aplicación y saltar a la venta de carga.

Algoritmo 1:

```
public function index(Request $request)
{
    $email = $request->input( key: 'email');
    $clave = $request->input( key: 'clave');
    if($email!=null and $clave!=null and Jugador::where( column: 'email', operator: '=', Input::get( key: 'email'))->exists() ){

        $login = DB::table('jugador')
            ->select('clave', 'api_token','id_jugador')
            ->where('email', '=', $email)
            ->get();
        $checkPass = Hash::check($clave, $login[0]->clave);
        if ($checkPass) {
            return response()->json(['api_token' => $login[0]->api_token, 'success' => true]);///,'img_perfil'=>URL_PERFIL]
        } else {
            return response()->json(['success' => false]);
        }
    }else{
        return response()->json(['success' => false]);
    }
}
```

Fuente: Omar Perez (2018)



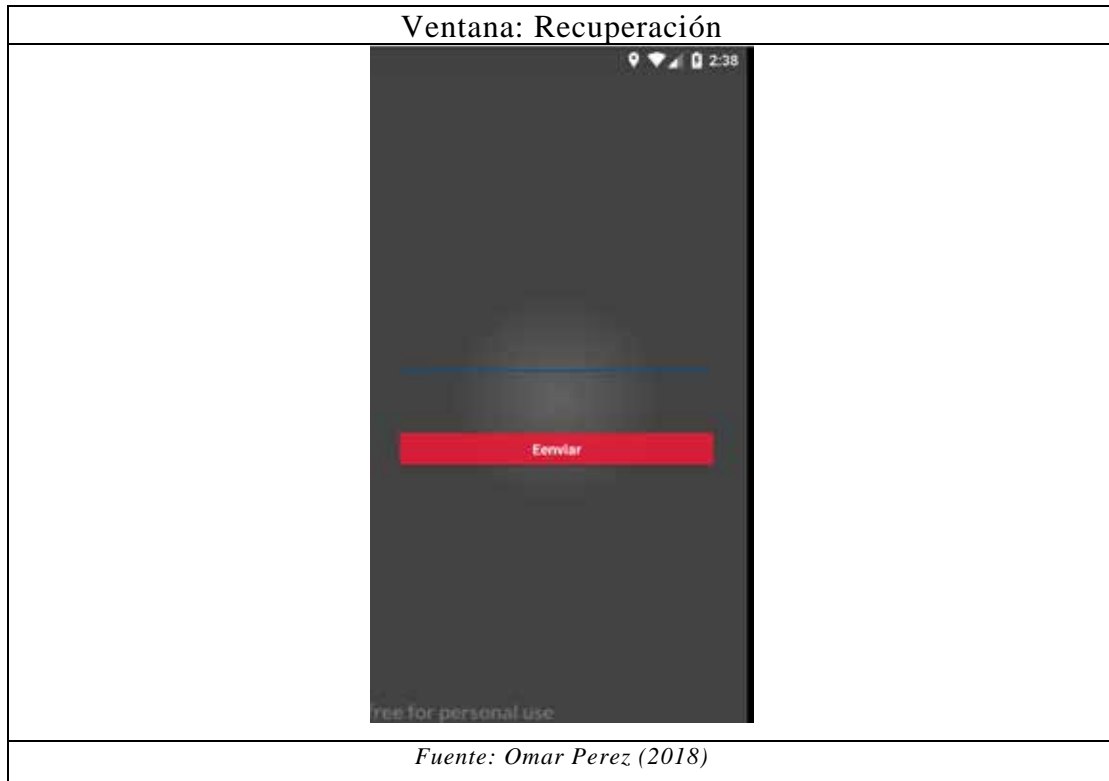
En la ventana de registro la aplicación valida los 5 campos según el tipo de dato, también el usuario escoge su sexo luego el botón de registro que tiene oyente de clic se pulsa para enviar la información al API la cual verifica mediante un algoritmo (ver algoritmo 2) si ya existe un usuario registrado con ese email y si los datos son correctos le notifica a la aplicación si no salta a la pantalla de carga.

Algoritmo 2:

```
public function create(Request $request)
{
    $jugador = new Jugador;
    if (Jugador::where( column: 'email', operator: '=', Input::get( key: 'email'))->exists()) {
        return response()->json(['success' => false]);
    } else {
        $jugador->id_estatus = '1';
        $jugador->clave = Hash::make($request->get( key: 'clave'));
        $jugador->email = $request->get( key: 'email');
        $jugador->nombre = $request->get( key: 'nombre');
        $jugador->sexo = $request->get( key: 'sexo');
        $jugador->fecha_nacimiento = $request->get( key: 'fecha_nacimiento');
        $jugador->api_token = str_random( length: 64);
        $jugador->tiene_imagen=0;
        $jugador->save();

        return response()->json(['api_token' => $jugador->api_token, 'success' => true]);
    }
}
```

Fuente: Omar Perez (2018)

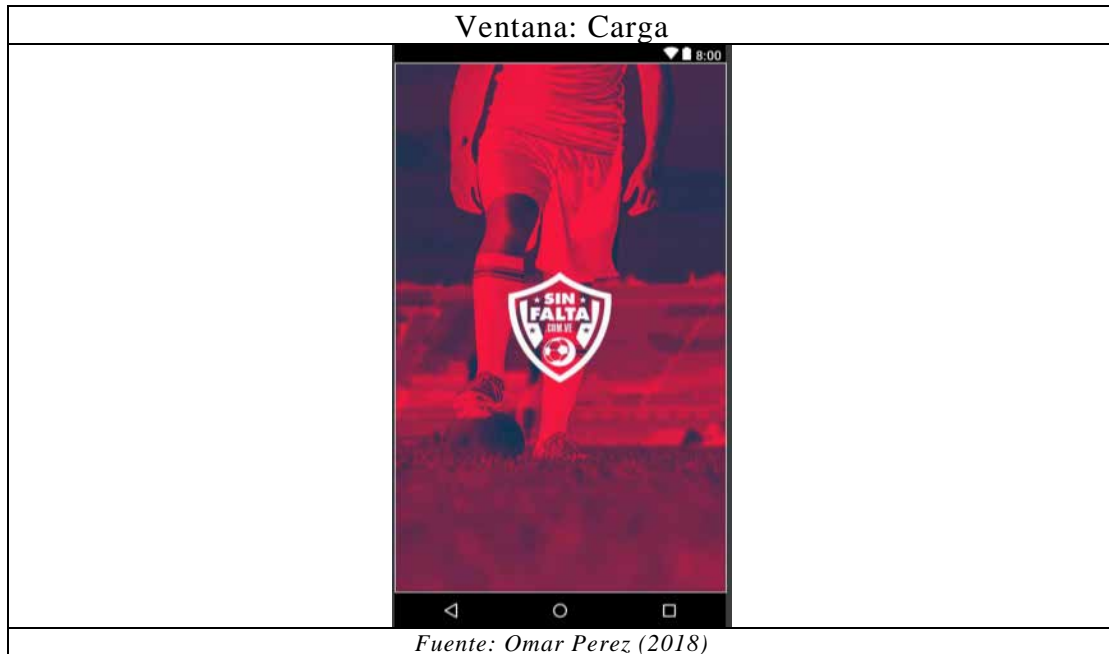


En la Ventana de recuperación es la más simple del programa solo se introduce el email validándolo el programa y al dar clic en el botón enviar que posee un oyente, se envía al API el cual verifica si existe el usuario mediante un algoritmo (ver algoritmo 3) y da respuesta a la aplicación, en caso de que exista el usuario a través de una función llamada PHPMail() envía un formato de recuperación de contraseña al email del usuario.

Algoritmo 3:

```
public function recuperar(Request $request)
{
    $jugador = new Jugador;
    if (Jugador::where( column: 'email', operator: '=', Input::get( key: 'email'))->exists()) {
        return response()->json(['success' => false]);
    } else {
        PHPMail();
        return response()->json(['estado' => 'Email enviado al correo', 'success' => true]);
    }
}
```

Fuente: Omar Perez (2018)



En la ventana de carga el usuario no introduce datos, sin embargo es una de las más importantes para el funcionamiento de la aplicación ya que esta se le pide al API datos necesarios para cada una de las ventanas siguientes. La Api consulta a la BD y da respuesta mediante un algoritmo (ver algoritmo 4) si se pudo cargar o no los datos para así guardarlos dentro del cache de la aplicación mediante la clase en java creada (ver clase 1).

Algoritmo 4:

```

public function index(Request $request) {
    $token=$request->get('key: 'api_token');
    $jugador = DB::table('jugador')
        ->select('id_jugador', 'tiene_imagen', 'peso')
        ->where('api_token', '=', $token)
        ->get();
    $tiene =DB::table('jugador_equipo')
        ->where( 'id_jugador', '=', $jugador[0]->id_jugador)
        ->select('id_jugador', 'id_equipo', 'id_rangoequipo')
        ->get();
    $tiene_entrenamiento =DB::table('jugador_entrenamiento')
        ->where( 'id_jugador', '=', $jugador[0]->id_jugador)
        ->exists();
    if($jugador[0]->tiene_imagen==1){...}else{...}
    if ($jugador[0]->peso or $jugador[0]->id_jugador ){...}else{...}
    if (!$tiene->isEmpty()){...}else{...}
}

```

Fuente: Omar Perez (2018)

Clase 1:

```
preferences=getApplicationContext().getSharedPreferences( name: "datos", Context.MODE_PRIVATE);
SharedPreferences.Editor editor=preferences.edit();
editor.putBoolean("tiene_entrenamiento", tiene_entrenamiento);
editor.putBoolean("tiene_equipo", tiene_equipo);
editor.putBoolean("tiene_imagen", tiene_imagen);
editor.putString("img_url", img_url+".png");
editor.apply();//comit

Intent intent=new Intent( packageContext: PantallaCargaActivity.this, AreaUsuarioActivity.class);
startActivity(intent);
```

Fuente: Omar Perez (2018)



Una vez que ya el usuario paso por el login y la pantalla de carga entra en el área principal de la aplicación, en este caso la pestaña de inicio en la cual se cargan las publicidades mediante una petición de token al Api y la mismas nos da el url de las imágenes y la descripción, cabe destacar que el montaje de las publicaciones es de manera manual por el administrador.

Ventana: Área de usuario/ Partidas



En esta pestaña se encuentra la Api proporcionada por GoogleMaps la cual se carga en un hilo diferente del programa y nos muestra las distintas partidas creadas en el mismo. Por el lado de los botones posee uno con el símbolo de ubicación el cual nos dirige a la nuestra localidad y el otro botón es para recargar los datos de la consultados con el token de sesión en la API para devolver los datos de las partidas mediante el algoritmo (ver algoritmo 5).

algoritmo 5:

```
public function show(Request $request)
{
    $partidas=DB::table('partida')
        ->where('id_estatus','=','1')
        ->select('id_partida','latitud','longitud','id_tipopartida')
        ->get();
    return response()->json(['partidas'=>$partidas,'success' => true]);
}
```

Fuente: Omar Perez (2018)

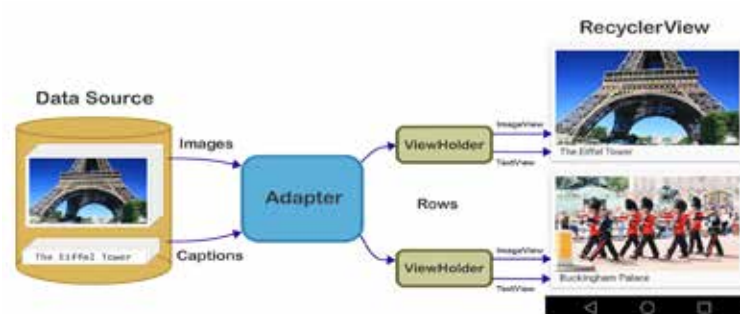
Ventana: Área de usuario/ Notificaciones



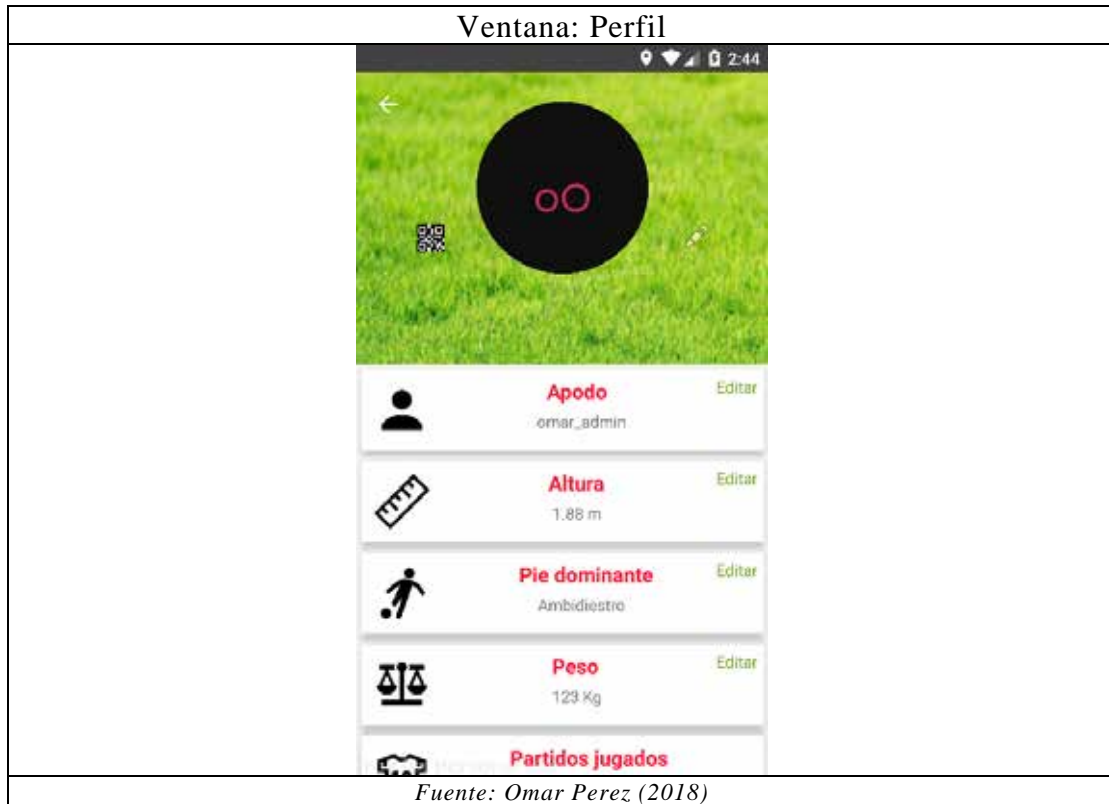
Fuente: Omar Perez (2018)

En la pestaña de notificaciones se envía el token de sesión y la respuesta de la API carga los datos. En el encontramos el primer elemento interesante, el cual es el recyclerview, este elemento es un listado de cartas donde la última carta pasa a ser la primera en el desplazamiento de la pantalla (ver Figura 1). Dentro del listamos encontramos las diferentes notificaciones en forma de cartas con las opciones que se pueden realizar de cara (aceptar/rechazar)

Figura 1:



Fuente: docs.microsoft.com (2018)



En el perfil se envía el token de sesión al API en el servidor y luego desvolver los datos del perfil del usuario mediante un algoritmo (ver algoritmo 6). En la imagen circular se usa la librería Glide la cual cargar las imágenes de una manera eficiente en la aplicación, además encontramos botones con oyentes los cuales el que tiene símbolo de QR genera la imagen de escaneo mediante una clase (ver clase 2) en formato QR mediante la librería Zxing para que cualquier usuario que lo escanee y pueda ser invitado a su lista de amigos, otro botón para editar la imagen de la aplicación y por último la el listado de los datos del usuario en un recyclerview de cartas en que cada una de ellas pueda ser editada.

Algoritmo 6:

```
public function show(Request $request)
{
    $buscar_otro=$request->get( key: "id_buscar");

    if(!$buscar_otro>0){
        $show = DB::table('jugador')
            ->select('api_token','apodo', 'altura', 'pie_dominante', 'peso','id_jugador', 'tiene_imagen')
            ->where('api_token', '=', $request->get( key: 'api_token'))
            ->first();
    }else{
        $show = DB::table('jugador')
            ->select('api_token','apodo', 'altura', 'pie_dominante', 'peso','id_jugador', 'tiene_imagen')
            ->where('id_jugador', '=', $request->get( key: "id_buscar"))
            ->first();
    }
}
```

Fuente: Omar Perez (2018)

Clase 2:

```
public class GenerateQRActivity extends AppCompatActivity {

    private ImageView ivQR;
    private ConstraintLayout constrainGrama;
    private ImageView imvFondo;
    private String datoQR, tipo_invitacion="";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_generate_qr);

        savedInstanceState = getIntent().getExtras();
        if(savedInstanceState != null) {
            datoQR= savedInstanceState.getString( key: "dato_qr");
            tipo_invitacion =savedInstanceState.getString( key: "tipo_invitacion");
        }

        ivQR=findViewById(R.id.ivQR);
        constrainGrama=findViewById(R.id.constrainGrama);

        MultiFormatWriter multiFormatWriter = new MultiFormatWriter();
        try {
            BitMatrix bitMatrix = multiFormatWriter.encode(datoQR, BarcodeFormat.QR_CODE, width: 200, height: 200);
            BarcodeEncoder barcodeEncoder = new BarcodeEncoder();
            Bitmap bitmap = barcodeEncoder.createBitmap(bitMatrix);
            ivQR.setImageBitmap(bitmap);
        } catch (WriterException e) {
            e.printStackTrace();
        }
    }
}
```

Fuente: Omar Perez (2018)



Por el lado de la venta de mis equipos se envía a la API el token de sesión y nos regresa los datos de los equipos a los que el usuario pertenece, en esta ventana se implementó la teoría de nodos y se creó una clase (ver clase 3), conjuntamente con una lista circular y su clase (ver clase 4), ya que se realizó una lista circular para poder cambiar entre los diferentes equipos que el usuario posee y así mostrar cada uno de los jugadores de esos equipos en una lista de cartas en las que se puede visualizar el perfil al darle clic en el icono del ojo de cada jugador. También posee los botones con oyentes correspondientes, por botón el código QR, el de ajustes y el de agregar usuarios al equipo.

Clase 3:

```
public class Nodo {
    private int valor;
    private Nodo siguiente;
    private Nodo anterior;

    public Nodo(int valor, Nodo anterior, Nodo siguiente) {
        this.valor = valor;
        this.siguiente = siguiente;
        this.anterior = anterior;
    }

    public int getValor() { return valor; }

    public void setValor(int valor) { this.valor = valor; }

    public Nodo getSiguiente() { return siguiente; }

    public void setSiguiente(Nodo siguiente) { this.siguiente = siguiente; }

    public Nodo getAnterior() { return anterior; }

    public void setAnterior(Nodo anterior) { this.anterior = anterior; }
}
```

Fuente: Omar Perez (2018)

Clase 4:

```
public class ListaCircular {

    private Nodo primero,ultimo,nodoCero;

    public ListaCircular() {
        this.primeros = null;
        this.ultimo = null;
    }

    public void ingresarNodo(int x){
        Nodo nuevo=new Nodo(x, anterior: null, siguiente: null);
        if (primero==null){
            primero=nuevo;
            ultimo=nuevo;
            primero.setSiguiente(primero);
            nuevo.setAnterior(ultimo);
            nodoCero=nuevo;
        }else{
            ultimo.setSiguiente(nuevo);
            nuevo.setSiguiente(primero);
            nuevo.setAnterior(ultimo);
            ultimo=nuevo;
            primero.setAnterior(ultimo);
        }
    }

    public Nodo getnodoCero() { return nodoCero; }
}
```

Fuente: Omar Perez (2018)



En la ventana de búsqueda de mis equipos se implementó una caja de texto la cual al escribir un algún nombre del equipo y al presionar el botón buscar el cual posee un oyente se envía el token de sesión al API para ejecutar el algoritmo de búsqueda (ver algoritmo 7), para luego retornarnos los diez (10) primeros equipos con sus datos al listado (recyclerview) de cartas, donde cada una de las mismas nos muestra las imágenes, nombre del equipo, la reputación, opción de ver el equipo y un botón con oyente para poder solicitar unirnos al equipo.

Algoritmo 7:

```

public function equipo(Request $request)
{
    if ($request) {
        $token=$request->get( 'key' 'api_token');
        $jugador = DB::table('jugador')
        ->select('id_jugador')
        ->where('api_token','=', $token)
        ->first();
        $query = trim($request->get( 'key' 'query'));
        $equipos = DB::table('equipo')
        ->select('id_equipo', 'nombre', 'reputacion_positiva', 'reputacion_negativa')
        ->where('equipo.nombre', 'LIKE', $query . '%')
        ->limit(10)
        ->get();
        return response()->json(['success' => true, 'equipos' => $equipos]);
    }else{
        return response()->json(['success' => false]);
    }
}

```

Fuente: Omar Perez (2018)



Por el lado de la ventana de mis amistades se le envía el token de sesión al Api y en respuesta nos carga en un listado de con los amigos que se tengan, conjuntamente cada una de ellas posee un oyente para ver el perfil de cada uno de los amigos. Además la lista de amistades cuenta con una clase (ver clase 5) en el adaptador del recyclerview que utiliza la clase filter() para filtrar las amistades a medida que se escribe el texto el buscado.

Clase 5

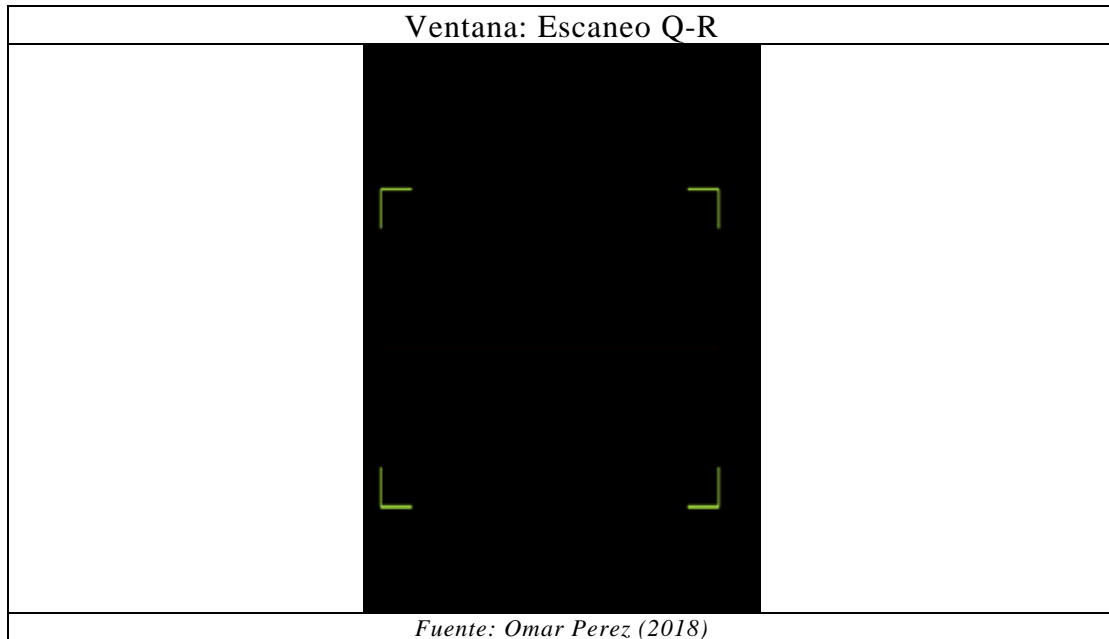
```

public Filter getFilter() {
    return new Filter() {
        @Override
        protected FilterResults performFiltering(CharSequence charSequence) {
            String charString = charSequence.toString();
            if (charString.isEmpty()) {
                mFilteredList = items;
            } else {
                ArrayList<PojoJugadores> filteredList = new ArrayList<>();
                for (PojoJugadores androidVersion : items) {
                    if (androidVersion.getApodo().toLowerCase().contains(charString) ||
                        androidVersion.getNombre().toLowerCase().contains(charString) ) {
                        filteredList.add(androidVersion);
                    }
                }
                mFilteredList = filteredList;
            }
            FilterResults filterResults = new FilterResults();
            filterResults.values = mFilteredList;
            return filterResults;
        }

        @Override
        protected void publishResults(CharSequence charSequence, FilterResults filterResults) {
            mFilteredList = (ArrayList<PojoJugadores>) filterResults.values;
            notifyDataSetChanged();
        }
    };
}

```

Fuente: Omar Perez (2018)



En la ventana de escaneo Q-R, utiliza la librería BarCodeScanner la cual nos crea por defecto una ventana en la cual utiliza la cámara del dispositivo para leer la información almacenada en el código que se desee escanear, dentro del código de la ventana implementa la clase `ZXingScannerView.ResultHandler()` para poder obtener el resultado del escaneo en una clase (ver Clase 6), para luego enviar la información junto con el token de sesión al Api y realizar la invitación al equipo o jugador.

Clase 6:

```
@Override
public void handleResult(Result rawResult) {
    id_scan =rawResult.getText();
    if (tipo_scan.equals("amigos")){
        enviarDatos1();
    }else{
        enviarDatos2();
    }
}
```

Fuente: Omar Perez (2018)

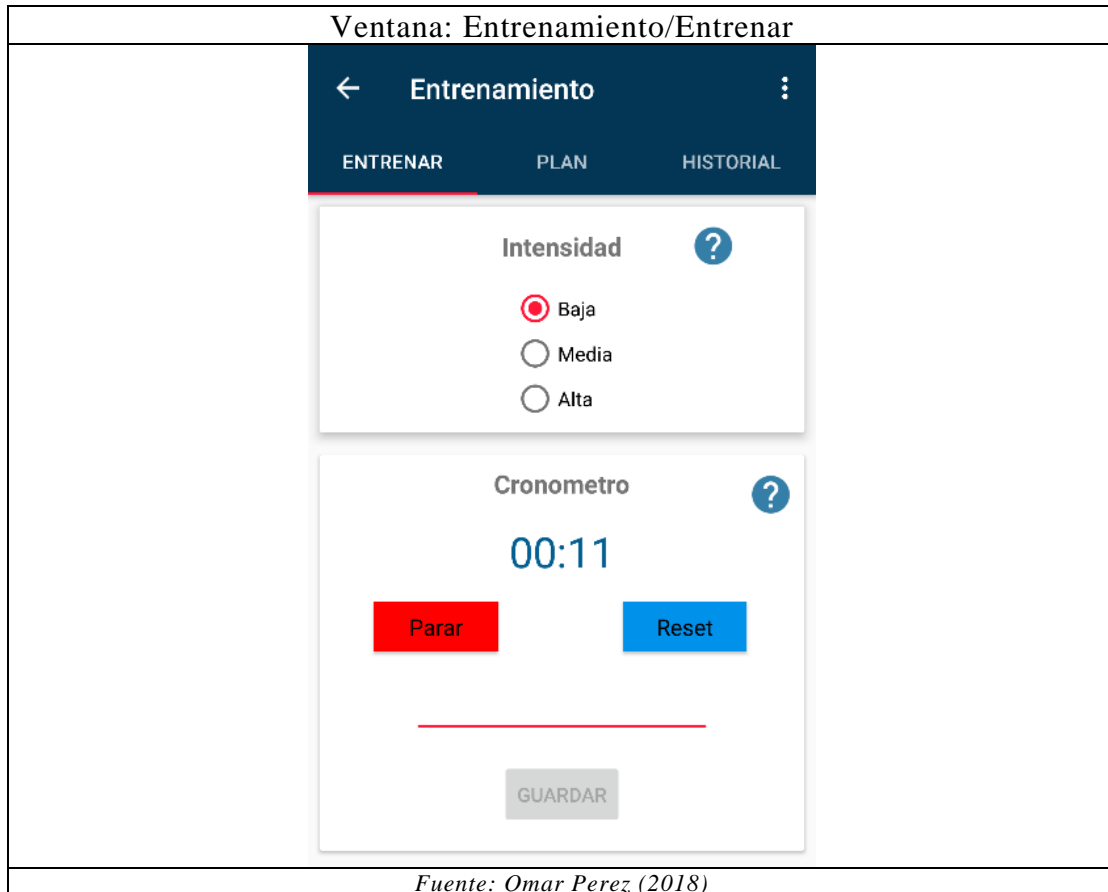


En la ventana de generar el código Q-R se implementó la librería Zxing la cual nos permite generar una imagen dentro de la aplicación con la información en código Q-R que se desea mostrar mediante un algoritmo (ver algoritmo 8). Primero se le pasa información por la clase intent() con el método putExtra() y cuando se recibe la información se crear una imagen Q-R de 200px de largo por ancho .

Algoritmo 8:

```
MultiFormatWriter multiFormatWriter = new MultiFormatWriter();
try {
    BitMatrix bitMatrix = multiFormatWriter.encode(datoQR, BarcodeFormat.QR_CODE, width: 200, height: 200);
    BarcodeEncoder barcodeEncoder = new BarcodeEncoder();
    Bitmap bitmap = barcodeEncoder.createBitmap(bitMatrix);
    ivQR.setImageBitmap(bitmap);
} catch (WriterException e) {
    e.printStackTrace();
}
```

Fuente: Omar Perez (2018)



En la ventana de entrenar encontramos la herramienta de quema de calorías, la cual mediante el algoritmo (ver algoritmo 9) que implementa las formulas establecidas en la fase I conjuntamente el cronometro elaborado con la clase Timer(), calcula la quema de calorías en un intervalo de tiempo, para después ser enviado los datos y el token de sesión de usuario al API para que se almacén en un historial.

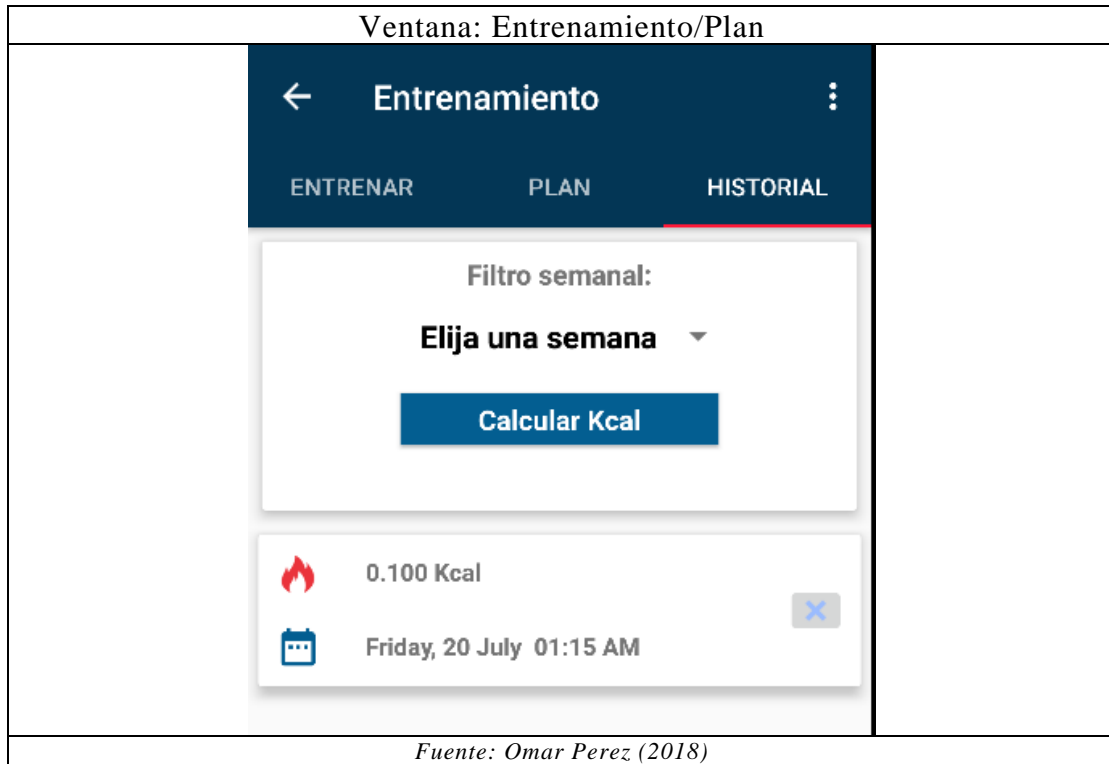
Algoritmo 9:

Algoritmo 9:

```
private void calculoCalorias() {
    DecimalFormat precision = new DecimalFormat( pattern: "0.000");
    double segundos = (SystemClock.elapsedRealtime() - cronometro.getBase())/1000;
    double horas =segundos/3600;
    calorias=MET*120*horas;
    etCalorias.setText(precision.format(calorias)+" Kcal");
}
```



Dentro de la ventana de plan se envía el token de sesión al API para obtener el tipo de plan de entrenamiento que la persona posee conjuntamente con una descripción elaborada de las distintas comidas en el día para llegar a la meta que se propuso el jugador. Cuenta con cartas para visualizar la información mejor e iconos correspondientes a cada tipo de elemento.



En el historial se carga la información enviando la sesión de token al API para luego mostrar la información en un listado de cartas con iconos, texto de las calorías, fecha y hora del entrenamiento y un botón con oyente para eliminar el entrenamiento, además cuenta con un filtro del listado por semanas que va desde la semana uno (1) hasta la semana cuatro (4) del mes.



Dentro del entrenamiento inicial el jugador debe colocar la altura en metros, el peso en kg, elegir los días que frecuenta hacer ejercicio a la semana y la meta que desea cumplir. Todo esto con el fin de dar inicio a su entrenamiento mediante las formulas establecidas en la fase I. Por otra parte los resultados inicial del IMC y TMB se mostraran en esta área una vez que el usuario realice clic en el botón guardar con oyente, el cual envía la información del entrenamiento inicial al servidor y la guarda.

4.1.12 Actividad IV: pruebas de caja negra y blanca dentro de la aplicación

Durante el desarrollo del código, es común por parte de los programadores el no contemplar ciertas situaciones que pueden generarse al ejecutar el programa. Es por esto que, se deben llevar a cabo pruebas unitarias de cada una de las funciones implementadas. Por consiguiente, se realizaron una serie pruebas de tipo caja blanca y caja negra a las pestañas más importantes dentro de la aplicación, cuya finalidad es detectar las posibles fallas que indiquen los resultados para poder erradicarlos. A éstas pruebas se le proporcionaron unos valores de entrada simulados para cada función, los cuales forman parte de los casos de prueba y son los que indican las salidas deseadas al final de la ejecución de las mismas.

- **Pruebas caja negra:**

La sigue es una prueba que consta de la entrada de cinco (5) valores el cual el usuario ingresa para ser enviados al API donde si el usuario no ha sido creado como lo es en este caso da una respuesta exitosa con un valor booleano “true”. Sin embargo se encontró que al almacenar el valor del correo se guarda incompleto en la base de datos esto es debido a que el campo correo en la tabla Jugadores cuenta con una longitud menor a la ingresada, por ende se procedió a modificar la amplitud de este campo a veinte y cinco (25).

Casos de prueba - registro				
Entradas		Salidas		
String	valor	Salida Esperada	Salida Obtenida	Resultado
String(25)	omar@gmail.com	Success=true	Success=true	Incompleto
String(20)	omar	Success=true	Success=true	OK
String(20)	perez	Success=true	Success=true	OK
String(20)	123456	Success=true	Success=true	OK
String(2)	26	Success=true	Success=true	OK

Fuente: Omar Perez (2018)

En esta prueba de escaneo de jugador por código Q-R se procedió a evaluar el resultado del escaneo por parte de la aplicación. En la cual mediante la cámara se lee una imagen suministrada. En este caso la imagen cuenta con el valor de un identificador con el numero cincuenta y seis (56), la salida esperada al programa es debería ser que ha sido éxito la lectura de la imagen e invitación del usuario. En cuanto al resultado final fue el esperado en esta prueba.

Casos de prueba – Escaneo jugador código Q-R				
Entradas		Salidas		
img	valor	Salida Esperada	Salida Obtenida	Resultado
String(bigInt)	Id=56	Success=true	Success=true	OK

Fuente: Omar Perez (2018)

Por otro lado se realizó una prueba en la pestaña enteramiento para verificar el comportamiento de la herramienta de quema de calorías. En la cual al ingresar una selección la cual posee un valor y al iniciar el cronometro establecido, nos da el tiempo y las Kcal quemadas en ese transcurso de tiempo, aparte de que al momento de inicio del cronometro se establece un modo para poder pararlo y reiniciarlo.

Casos de prueba – Entrenamiento				
Entradas		Salidas		
entrenamiento	valor	Salida Esperada	Salida Obtenida	Resultado
Integer(1)	1.2	String(time)	1:23, Kcal	OK
clic	true	Bolean	false	OK

Fuente: Omar Perez (2018)

- **Prueba de caja blanca:**

En lo referente al módulo de entrenamiento, se probó la función de corrección de cálculo de calorías `calculoCalorias()`, la cual se enfocó en el resultado que daba el dato segundos al pasar tres (3) valores esenciales: uno igual a cero y los dos otros mayor y menor a cero. Al ejecutar la prueba, se encontró con un error de división entre cero, que se originó de la división realizada para obtener el valor correspondiente al tiempo en segundos, y el error al ser menor que cero se originó un valor erróneo ya que el tiempo no puede ser negativo. En base a estas fallas, se procedió a implementar una condición de valor mayor a cero.

Casos de prueba - <code>calculoCalorias()</code>				
Entradas		Salidas		
segundos	gamma	Salida Esperada	Salida Obtenida	Resultado
double	0.0	double	ZeroDivisionError	Falla
double	1.2	double	double	OK
double	-1.2	double	double	Respuesta errónea

Fuente: Omar Perez (2018)

Se procedió a realizar una prueba en el módulo de equipos en donde se muestra la reputación en una escala de cinco (5) estrellas donde cero (0) estrellas es el menor de los resultados posibles y cinco (5) el mayor. La prueba consistió el introducir 2 parámetros para los posibles caminos de ejecución, los cuales son la reputación positiva y negativa del equipo para poder realizar el cálculo estadístico de las estrellas.

Casos de prueba - calcularReputacion()				
Entradas		Salidas		
reputaciones	gamma	Salida Esperada	Salida Obtenida	Resultado
Integer, Integer	1, 2	0	0	Ok
Integer, Integer	2, 1	1	1	OK
Integer, Integer	0, 1	0	0	OK
Integer, Integer	10, 5	5	5	OK

Fuente: Omar Perez (2018)

En la prueba de guardar los datos para el entrenamiento del jugador, se realizó en el API la prueba en la función guardarEntIni (), en donde consiste en calcular los datos de TMB e IMC. La prueba consistió en introducir los datos de altura, peso, y facto de sexo con variantes de tipos de datos en cuanto con comas o sin comas y mayores o menores, lo resultados fueron los esperados por el programa.

Casos de prueba - guardarEntIni ()				
Entradas		Salidas		
reputaciones	gamma	Salida Esperada	Salida Obtenida	Resultado
Integer, Double, Double,	1, 1.88, 120.3	IMC =34.03, TMB=2500	IMC =34.03, TMB=2500	Ok
Integer, Double, Double,	1, 1.70, 90.3	IMC =31.24, TMB =2275	IMC =31.24, TMB =2275	OK
Integer, Double, Double	1, 2, 100	IMC =25.00, TMB =1800	IMC =25.00, TMB =1800	OK

Fuente: Omar Perez (2018)

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

En el presente trabajo de investigación se desarrolló un software para el control de los entrenamientos de los jugadores, el cual además cuenta con características para que los mismo puedan organizar sus partidas, tener amistades, equipos y perfiles. Es importante resaltar que se implementó nuevas tecnologías como lo son el escaneo de códigos Q-R y la geo localización. Consecuente con lo antes mencionado se sentaron las bases de un área de investigación poco explorada en la Universidad José Antonio Páez, en las cuales futuros proyectos pudiesen basar su estudio utilizando este trabajo como antecedente.

Por otro lado, se diagnosticó la situación actual de los jugadores de futbol y sus prioridades dentro del entrenamiento, gracias a la interacción y colaboración de los mismos, conjuntamente con las entidades encargadas de prestar servicio a este deporte. Se observó todos los obstáculos que presentan los jugadores que practican el futbol, tanto en sus entrenamientos, como el apoyo necesario para lograr el éxito en este deporte altamente competitivo.

Seguidamente, se establecieron los requerimientos funcionales y no funcionales, en donde se consultó por medio de estadísticas la aceptación y rechazo de algunas funcionalidades planteadas previamente, las cuales arrojaron resultados que determinaron ciertos cambios necesarios. Por ende, gracias al uso de una metodología ágil, permitió que el desarrollo del software se adaptase con facilidad a los cambios de requerimientos que surgieron a través de las encuestas mencionadas, los cuales contribuyeron a complementar las especificaciones del sistema, ajustándolo a las necesidades del jugador.

Consecuentemente, se realizaron esquemas de modelo para las distintas partes de la aplicación. En donde se implementaron los diagramas UML referentes al API, al modelo de base de datos, y distintos estados dentro de la aplicación, lo que permitió

generar una idea de la estructura y secuencia que se tuvo que implementar para la construcción de la aplicación.

Además, se analizaron los métodos y técnicas existentes en la librería Zxing y BarCode orientados a generar y escanear imágenes de código Q-R, se revisó la documentación del API GoogleMaps y se estudió la correlación entre el ejercicio, tiempo y quema de calorías. Todo con el fin de implementar lo antes mencionado dentro de la aplicación.

Partiendo por las librerías Zxing y BarCode se evidenció que su implementación fue de manera sencilla, ya que para poder escanear el código solo se necesitan los permisos de usuario para la activación de la cámara, y por la parte de generar el código con solo crear un objeto de esta librería y pasar los parámetros de información, y dimensiones de la imagen se genera automáticamente. En cuanto al Api de GoogleMaps lo más tedioso fue generar y configurar la llave de permiso generada por la consola de Google. Cabe destacar que la implementación de la herramienta de quema de calorías en la aplicación fue un trabajo más tedioso, ya que existen varios factores que intervienen en la quema de calorías y el tipo de meta a la cual se desea llegar con el entrenamiento de los jugadores.

Con el motivo de asegurar que el programa de software no contenga errores a la hora de ejecución, se verificaron los algoritmos que se implementaron en las nuevas tecnologías. Para esto, las pruebas de caja blanca y caja negra ayudaron a descubrir fallas no contempladas y contribuyeron a la mejora del código fuente del sistema. Para así tener un software final de calidad.

5.2 Recomendaciones

Dentro de la realización de un proyecto tan grande y ambicioso como lo fue éste, siempre se desea que exista una mejora continua del mismo, por lo tanto, con el fin de mantener la escalabilidad del software y ampliar el alcance de la investigación, se conciben las siguientes recomendaciones:

- Desarrollar una ventana propia para modificación y carga de imágenes dentro de la aplicación con el fin de no depende de una librería y poder modificarla a las necesidades previstas.
- Vincular el poder compartir de información entre distintas plataformas.
- Mejorar la interfaz gráfica de la aplicación para que compita visualmente con las aplicaciones más destacadas en diseño del mercado.
- Implementar más funcionalidades que permita mejorar los resultados de los jugadores en la herramienta de control de enteramientos.
- Establecer un plan de mercadeo para aumentar las descargas dentro de la PlayStore.
- Mejorar los algoritmos de búsqueda de jugadores y equipos para sugerir las posibles opciones a la hora de seleccionar.
- Dedicar el diseño de la aplicación multiplataforma en un lenguaje nativo por parte del SO IOS.
- Incluir funcionalidades a la aplicación para los árbitros.

REFERENCIAS

Bibliografía

- A., J. M. (2016). **Desarrollo De Una Aplicación Móvil Sobre La Plataforma Android Para La Liga De Futbol Sala De La Facultad De Ciencias De La Universidad Central De Venezuela.** Obtenido de http://saber.ucv.ve/bitstream/123456789/14698/1/TEG_JoseAlmeida-JoseRomero.pdf
- Alicante, L. U. (2007). Obtenido de <https://rua.ua.es/dspace/bitstream/10045/4412/5/03c-AplicacionesWeb.pdf>
- B., J. (2014). **Análisis De Los Desplazamientos A Muy Alta Velocidad En Fútbol Profesional Mediante Tecnología GPS.** Obtenido de https://rio.upo.es/xmlui/bitstream/handle/10433/1539/fco.%20javier_toscano_tesis.pdf?sequence=1
- Beck, K. (1999). **Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP).** Obtenido de <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
- Beck, K. (2005). **Extreme Programming Explained. Second Edition.** Obtenido de <http://ptgmedia.pearsoncmg.com/images/9780321278654/samplepages/9780321278654.pdf>
- Benedict, H. (s.f.). http://www.healthfitonline.com/resources/harris_benedict.php.
- compendiumofphysicalactivities.** (s.f.). <https://sites.google.com/site/compendiumofphysicalactivities/compendia>
- .
- .

Base de datos. (s.f.).

<https://users.dcc.uchile.cl/~cguierr/cursos/BD/extendido.pdf>.

E., O. J. (2015). **Seguridad En Dispositivos Móviles Android**. Obtenido de <http://stadium.unad.edu.co/preview/UNAD.php?url=/bitstream/10596/3614/1/59836994.pdf>

G., A. (2006). **El Proyecto de Investigación**. Obtenido de <https://evidencia.com/wp-content/uploads/2014/12/EL-PROYECTO-DE-INVESTIGACION-6ta-Ed.-FIDIAS-G.-ARIAS.pdf>

G., P. (2013). **Desarrollo De Una Aplicación Generadora Y Lectora De Códigos QR Seguros En Android** . Obtenido de <https://e-archivo.uc3m.es/bitstream/handle/10016/19043/TFG-Carlos-Pliego-Garcia.pdf>

Goretty, N. T. (2011). **MySQL**. Obtenido de <http://www.gridmorelos.uaem.mx/~mcruz/cursos/miic/MySQL.pdf>

Group, O. M. (s.f.). <https://www.omg.org>.

JAVA. (s.f.). Obtenido de

https://www.java.com/es/download/faq/whatis_java.xml

Libertador, U. P. (2006). **Manual de Trabajos de Grado de Especialización y Maestría y Tesis Doctorales. Caracas: FEDUPEL**. Obtenido de <http://neutron.ing.ucv.ve/NormasUPEL2006.pdf>

M, S. P. (2010). **Tipos y diseño de la investigación**. Obtenido de <http://planificaciondeproyectosemirarismendi.blogspot.com/>

mockups. (s.f.). <http://estudioka.es/que-es-un-mock-up/>.

MVC. (s.f.).

https://www.ibm.com/support/knowledgecenter/es/SSZLC2_8.0.0/com.ibm.commerce.developer.doc/concepts/csdmvcdespat.html.

otros, M. A. (2009). **Manual de PHP**. Obtenido de <http://php.net/manual/es/index.php>

P., F. (2017). **NUTFIT, Herramienta Para Motivar Buenos Hábitos Alimenticios**. Obtenido de <http://132.248.52.100:8080/xmlui/bitstream/handle/132.248.52.100/13748/TESIS.pdf?sequence=1>

peso, **Perdida de peso**. (s.f.). <https://www.menshealth.es/nutricion/dietas/articulo/La-dieta-para-acelerar-tu-metabolismo#pagina-4>.

Proteina. (s.f.). <http://www.scielo.org.ve/pdf/alan/v63n4/art04.pdf>.

proteina, t. (s.f.). <https://www.vitonica.com/dietas/necesidades-proteicas-segun-deportista>.

resistencia, D. p. (s.f.). <https://www.abc.es/salud/especialistas/20140520/abci-dieta-futbolista-201405131258.html>.

Riehle, D. (. (2000). **Framework Design: A Role Modeling Approach**. Obtenido de <http://dirkriehle.com/computer-science/research/dissertation/diss-a4.pdf>

salud, o. m. (s.f.). <http://www.who.int/es>.

Silberschatz, A. K. (s.f.). **SISTEMA OPERATIVO ANDROID: CARACTERÍSTICAS Y FUNCIONALIDAD PARA DISPOSITIVOS MÓVILES**. Obtenido de <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/2687/0053M722.pdf;jsessionid=67E99C88CA73191AB3FDC2BA3CEA6C3B?sequence=1>

Spencer, T. (2014). **No REST for the whippet**. Obtenido de <https://fiznool.com/blog/2014/09/16/no-rest-for-the-whippet/>

ANEXO A

Instrumentos de investigación



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN**

Encuesta

Objetivo: Conocer la opinión y situación de las personas respecto a la práctica del deporte, el uso de herramientas de entrenamiento y condiciones de salud para el desarrolladores del software, de tal manera que se pueda construir un sistema que cumpla con los objetivos de la investigación.

1. Usted practica frecuentemente deporte

Si

No

2. Considera que su salud es óptima.

Si

No

3. Cree usted que está en su peso ideal.

Si

No

4. Usa alguna herramienta para manejar sus resultados en su entrenamiento.

Si

No

5. Le gusta realizar ejercicio en grupo.

Si

No

6. Sufre de alguna condición de salud.

Si

No

7. Usted usaria alguna aplicación para tener un control de sus entrenamientos.

Si

No

ANEXO B

CODE	METS	MAJOR HEADING	SPECIFIC ACTIVITIES
11770	1.3	occupation	typing, electric, manual or computer
11780	6.3	occupation	using heavy power tools such as pneumatic tools (e.g., jackhammers, drills)
11790	8.0	occupation	using heavy tools (not power) such as shovel, pick, tunnel bar, spade
11791	2.0	occupation	walking on job, less than 2.0 mph, very slow speed, in office or lab area
11792	3.5	occupation	walking on job, 3.0 mph, in office, moderate speed, not carrying anything
11793	4.3	occupation	walking on job, 3.5 mph, in office, brisk speed, not carrying anything
11795	3.5	occupation	walking on job, 2.5 mph, slow speed and carrying light objects less than 25 lbs
11796	3.0	occupation	walking, gathering things at work, ready to leave
11797	3.8	occupation	walking, 2.5 mph, slow speed, carrying heavy objects more than 25 lbs
11800	4.5	occupation	walking, 3.0 mph, moderately and carrying light objects less than 25 lbs
11805	3.5	occupation	walking, pushing a wheelchair
11810	4.8	occupation	walking, 3.5 mph, briskly and carrying objects less than 25 lbs
11820	5.0	occupation	walking or walk downstairs or standing, carrying objects about 25 to 49 lbs
11830	6.5	occupation	walking or walk downstairs or standing, carrying objects about 50 to 74 lbs
11840	7.5	occupation	walking or walk downstairs or standing, carrying objects about 75 to 99 lbs
11850	8.5	occupation	walking or walk downstairs or standing, carrying objects about 100 lbs or more
11870	3.0	occupation	working in scene shop, theater actor, backstage employee
12010	6.0	running	jog/walk combination (jogging component of less than 10 minutes) (Taylor Code 180)
12020	7.0	running	jogging, general
12025	8.0	running	jogging, in place
12027	4.5	running	jogging, on a mini-tramp
12029	6.0	running	Running, 4 mph (13 min/mile)
12030	8.3	running	running, 5 mph (12 min/mile)
12040	9.0	running	running, 5.2 mph (11.5 min/mile)
12050	9.8	running	running, 6 mph (10 min/mile)
12060	10.5	running	running, 6.7 mph (9 min/mile)
12070	11.0	running	running, 7 mph (8.5 min/mile)
12080	11.5	running	running, 7.5 mph (8 min/mile)
12090	11.8	running	running, 8 mph (7.5 min/mile)
12100	12.3	running	running, 8.6 mph (7 min/mile)
12110	12.8	running	running, 9 mph (6.5 min/mile)
12120	14.5	running	running, 10 mph (6 min/mile)
12130	16.0	running	running, 11 mph (5.5 min/mile)
12132	19.0	running	running, 12 mph (5 min/mile)
12134	19.8	running	running, 13 mph (4.6 min/mile)
12135	23.0	running	running, 14 mph (4.3 min/mile)
12140	9.0	running	running, cross country
12150	8.0	running	running, (Taylor code 200)
12170	15.0	running	running, stairs, up
12180	10.0	running	running, on a track, team practice
12190	8.0	running	running, training, pushing a wheelchair or baby carrier
12200	13.3	running	running, marathon
13000	2.3	self care	getting ready for bed, general, standing
13009	1.8	self care	sitting on toilet, eliminating while standing or squatting
13010	1.5	self care	bathing, sitting
13020	2.5	self care	dressing, undressing, standing or sitting
13030	1.5	self care	eating, sitting
13035	2.0	self care	talking and eating or eating only, standing
13036	1.5	self care	taking medication, sitting or standing
13040	2.0	self care	grooming, washing hands, shaving, brushing teeth, putting on make-up, sitting or standing
13045	2.5	self care	hairstyling, standing

CODE	METS	MAJOR HEADING	SPECIFIC ACTIVITIES
13046	1.3	self care	having hair or nails done by someone else, sitting
13050	2.0	self care	showering, toweling off, standing
14010	2.8	sexual activity	active, vigorous effort
14020	1.8	sexual activity	general, moderate effort
14030	1.3	sexual activity	passive, light effort, kissing, hugging
15000	5.5	sports	Alaska Native Games, Eskimo Olympics, general
15010	4.3	sports	archery, non-hunting
15020	7.0	sports	badminton, competitive (Taylor Code 450)
15030	5.5	sports	badminton, social singles and doubles, general
15040	8.0	sports	basketball, game (Taylor Code 490)
15050	6.0	sports	basketball, non-game, general (Taylor Code 480)
15055	6.5	sports	basketball, general
15060	7.0	sports	basketball, officiating (Taylor Code 500)
15070	4.5	sports	basketball, shooting baskets
15072	9.3	sports	basketball, drills, practice
15075	7.8	sports	basketball, wheelchair
15080	2.5	sports	billiards
15090	3.0	sports	bowling (Taylor Code 390)
15092	3.8	sports	bowling, indoor, bowling alley
15100	12.8	sports	boxing, in ring, general
15110	5.5	sports	boxing, punching bag
15120	7.8	sports	boxing, sparring
15130	7.0	sports	broomball
15135	5.8	sports	children's games, adults playing (e.g., hopscotch, 4-square, dodg)
15138	6.0	sports	cheerleading, gymnastic moves, competitive
15140	4.0	sports	coaching, football, soccer, basketball, baseball, swimming, etc.
15142	8.0	sports	coaching, actively playing sport with players
15150	4.8	sports	cricket, batting, bowling, fielding
15160	3.3	sports	croquet
15170	4.0	sports	curling
15180	2.5	sports	darts, wall or lawn
15190	6.0	sports	drag racing, pushing or driving a car
15192	8.5	sports	auto racing, open wheel
15200	6.0	sports	fencing
15210	8.0	sports	football, competitive
15230	8.0	sports	football, touch, flag, general (Taylor Code 510)
15232	4.0	sports	football, touch, flag, light effort
15235	2.5	sports	football or baseball, playing catch
15240	3.0	sports	frisbee playing, general
15250	8.0	sports	frisbee, ultimate
15255	4.8	sports	golf, general
15265	4.3	sports	golf, walking, carrying clubs
15270	3.0	sports	golf, miniature, driving range
15285	5.3	sports	golf, walking, pulling clubs
15290	3.5	sports	golf, using power cart (Taylor Code 070)
15300	3.8	sports	gymnastics, general
15310	4.0	sports	hacky sack
15320	12.0	sports	handball, general (Taylor Code 520)
15330	8.0	sports	handball, team
15335	4.0	sports	high ropes course, multiple elements
15340	3.5	sports	hang gliding

ANEXO C-1

Documento de definición de requerimientos

Nombre del proyecto	desarrollo de una aplicación multiplataforma para el control del desempeño del deportista
Descripción	Se desarrollará un software que permita a los deportistas o personas que desean practicar algún deporte poder organizar partidas de fútbol para ejercitarse.
Fecha de inicio	03/12/2017
Versión	1.0

Definición Requerimientos

1	Construir una interfaz gráfica para el manejo del software
2	Organizar partidas y equipos
3	Subir imágenes de perfil y datos de los mismos usuarios
4	Agregar filtros para buscar jugadores y equipos
5	Mostrar las partidas mediante geo-localización con la api de googleMaps
6	Poder guardar los entrenamientos y sus quemaduras de calorías en historiales
7	Escanear códigos QR para facilitar las acciones de invitar de los jugadores
8	Crear un plan con sugerencias para que los deportistas cumplan las metas escogidas

ANEXO C-2

Documento de especificación de requerimientos

Nombre del proyecto	desarrollo de una aplicación multiplataforma para el control del desempeño del deportista
Descripción	Se desarrollará un software que permita a los deportistas o personas que desean practicar algún deporte poder organizar partidas de fútbol para ejercitarse.
Fecha de inicio	03/12/2017
Versión	1.0

Especificación de requerimientos

- 1) **Construir una interfaz gráfica para el manejo del software**
 - a) Desarrollar las ventanas de la interfaz
 - i) Un área de usuario principal
 - ii) Un Perfil para los jugadores
 - iii) Un filtro de búsqueda para encontrar jugadores y equipos
 - iv) Un área de entrenamiento para los deportistas
 - v) Una sección para escaneo de códigos Q-R
 - vi) Una sección para manejar los equipos
 - b) Agregar los botones de la interfaz
 - i) Implementar Botones para las distintas acciones de la aplicación, que correspondan a la funcionalidad
 - c) Agregar iconos de la interfaz
 - i) Implementar iconos los cuales ayuden a la aplicación a ser más fácil de usar a los usuarios
- 2) **Organizar partidas y equipos**
 - a) Mostrar los equipos
 - b) Mostrar los jugadores del equipo
 - c) Poder ver el perfil de cada jugador en el equipo
 - d) Mostrar el tipo de equipo

- e) Eliminar o salirse del equipo
- f) Modificar los datos del equipo
- 3) Subir imágenes de perfil y datos de los mismos usuarios
 - a) Transformar la imagen en cadena de texto
 - b) Convertir la imagen en formato .jpg
 - c) Subir imagen al servidor
- 4) Agregar filtros para buscar jugadores y equipos
 - a) Implementar algoritmo de búsqueda de jugadores o equipo en la API
 - b) Optimizar la representación de la información a buscar los jugadores o equipo
- 5) Mostrar las partidas mediante geo-localización con la api de GoogleMaps
 - a) Crear la llave de GoogleMaps
 - b) Implementar la api de GoogleMaps
 - c) Cambiar los iconos de marcas en el mapa
 - d) Modificar el estilo del mapa
 - e) Usar hilos para la carga de información en el mapa
- 6) Poder guardar los entrenamientos y sus quemas de calorías en historiales
 - a) Construir un cronometro
 - b) Implementar el algoritmo de quema de calorías
 - c) Guardar los entrenamientos en el servidor
 - d) Mostrar el historial de los entrenamientos
- 7) Crear un plan con sugerencias para que los deportistas cumplan las metas escogidas
 - a) Elegir el tipo de entrenamiento que desea realizar
 - b) Establecer las sugerencias del tipo de entrenamiento
 - c) Mostrar el plan seleccionado