

CIS2210CP16

Fantasy Walker

Luis Manuel Peñaranda Ramírez

Juan Camilo Merchán Loza

Nicolas Daniel Vargas Ortiz

PONTIFICIA UNIVERSIDAD JAVERIANA

FACULTAD DE INGENIERIA

SYSTEMS ENGINEERING PROGRAM

BOGOTÁ, D.C.

2023

CIS2210CP16

Fantasy Walker

Autor(es):

Luis Manuel Peñaranda Ramírez

Juan Camilo Merchán Loza

Nicolas Daniel Vargas Ortiz

MEMORIA DE PROYECTO FINAL REALIZADO CON EL FIN DE CUMPLIR
UNO DE LOS REQUISITOS DEL TITULO DE INGENIERIA DE SISTEMAS

Director

Ing. Leonardo Florez Valencia

Jurados del proyecto final de pregrado

Ing. Efraín Ortiz Pabón

Ing. Luis Carlos Diaz Chaparro

PONTIFICIA UNIVERSIDAD JAVERIANA

FACULTAD DE INGENIERIA

SYSTEMS ENGINEERING PROGRAM

BOGOTÁ, D.C.

Mayo, 2023

**PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERIA DE SISTEMAS**

Rector de la Pontificia Universidad Javeriana

Luis Fernando Múnera S.J.

Decano de la facultad de ingeniería

Ing. Lope Hugo Barrero Solano

Director del departamento de ingeniería de sistemas

Ing. Cesar Julio Bustacara Medina

Director de la facultad de ingeniería de sistemas

Ing. Carlos Andrés Parra Acevedo

Artículo 23 de la Resolución No. 1 de junio de 1946

“La Universidad no se hace responsable de los conceptos emitidos por sus alumnos en sus proyectos de grado. Sólo velará porque no se publique nada contrario al dogma y la moral católica y porque no contengan ataques o polémicas puramente personales. Antes bien, que se vean en ellos el anhelo de buscar la verdad y la Justicia”

AGRADECIMIENTOS

El presente trabajo de grado se realizó gracias al apoyo incondicional y compañía de nuestras familias, profesores y amigos, en especial, queremos agradecerle a: Ing. Leonardo Florez Valencia, Juan Pablo Vera Fuentes, Diana Carvajal, Andrés Felipe Merchán Loza, Angelica María Loza Castillo, Horacio Alberto Merchán Paredes, Alexandra Ortiz, Mauricio Vargas y Diana Marcela Peñaranda Ramírez.

Contenido

1 RESUMEN	9
1.1 INTRODUCCIÓN	9
2 DESCRIPCIÓN GENERAL.....	10
2.1 ANTECEDENTES, PROBLEMA Y SOLUCIÓN PROPUESTA.....	10
2.1.1 Descripción de la problemática u oportunidad	10
2.1.2 Formulación del problema.....	10
2.1.3 Propuesta de solución.....	10
2.1.4 Justificación de la solución.....	11
2.2 OBJETIVOS	11
2.2.1 Objetivo general.....	11
2.2.2 Objetivos Específicos	11
2.3 ENTREGABLES, ESTÁNDARES UTILIZADOS Y JUSTIFICACIÓN.....	11
3 MARCO TEÓRICO	13
3.1 FUNDAMENTOS Y CONCEPTOS RELEVANTES PARA EL PROYECTO.	13
3.2 ANÁLISIS DE ALTERNATIVAS DE SOLUCIÓN	14
3.2.1 Alternativas de solución e impacto	14
3.2.2 Comparación de alternativas.....	16
4 ANÁLISIS DEL PROBLEMA	18
4.1 REQUERIMIENTOS	18
5 ANÁLISIS DE IMPACTO	21
6 DESCRIPCIÓN GLOBAL	22
6.1 PERSPECTIVA DEL PRODUCTO.....	22
6.1.1 Funciones del producto.....	25
6.1.2 Características de usuario	25
6.2 SUPUESTOS Y RESTRICCIONES DEL PRODUCTO	28
6.2.1 Supuestos.....	28
6.2.2 Restricciones de Software	28
6.2.3 Restricciones de hardware.....	28
6.3 REQUERIMIENTOS ESPECÍFICOS	29
6.3.1 Interfaces con el usuario.....	29
6.3.2 Interfaces con el sistema.....	30
6.3.3 Atributos de calidad.....	30
6.4 ARQUITECTURA 4 + 1	32
6.4.1 Descripción General del 4 + 1	33
6.4.2 Vista Lógica 4 + 1.....	34

6.4.3 Vista de Desarrollo 4 + 1	34
6.4.4 Vista de Procesos 4 + 1	35
6.4.5 Vista Física 4 + 1	36
6.4.6 Historia de usuarios	37
6.4.7 Escenarios 4 + 1	38
7 PROCESO	40
7.1 FASE METODOLÓGICA 1 (DISEÑO E IMPLEMENTACIÓN DE ELEMENTOS BÁSICOS DE LA APLICACIÓN)	40
7.1.1 Método	40
7.1.2 Actividades	40
7.1.3 Resultados esperados	41
7.2 FASE METODOLÓGICA 2 (DISEÑO E IMPLEMENTACIÓN DE FUNCIONALIDAD DE EJERCICIO)	41
7.2.1 Método	41
7.2.2 Actividades	41
7.2.3 Resultados esperados	41
7.3 FASE METODOLÓGICA 3 (ELEMENTOS AUDIOVISUALES Y DE GAMIFICACIÓN)	42
7.3.1 Método	42
7.3.2 Actividades	42
7.3.3 Resultados esperados	42
8 DESARROLLO DE LA SOLUCIÓN	43
8.1 DESARROLLO Y FUNCIONAMIENTO DEL <i>BACKEND</i>	43
8.2 DESARROLLO DE LA APLICACIÓN MÓVIL	44
8.3 CREACIÓN DE CONTENIDO Y EMBELLECIMIENTO DE LAS PANTALLAS	45
8.4 TAREAS Y MANEJO DE RAMAS Y <i>COMMITTS</i>	45
9 RESULTADOS	47
9.1 APLICACIÓN	47
9.1.1 Implementación	47
9.2 PRUEBAS	49
9.2.1 Metodología elegida	49
9.2.2 Implementación de la metodología	49
9.2.3 Resultados de pruebas	51
10 COMPROMISO DE APOYO DE LA INSTITUCIÓN	55
11 DERECHOS PATRIMONIALES	56
12 CONCLUSIONES	57
13 REFERENCIAS	58

14 ANEXOS.....	60
-----------------------	-----------

1 Resumen

Hace dos años la humanidad se enfrentó a una crisis sin precedentes que diezmo la población mundial y nos obligó a permanecer encerrados en nuestros hogares por al menos dos años, generando que un problema, que ya era común en nuestra sociedad, se volviera uno aún más grave: el sedentarismo. En este documento planteamos la ejecución de una solución que propone trabajar el sedentarismo en poblaciones jóvenes mediante el uso de fantasía lúdica y elementos de videojuegos con el fin de generar una cultura activa y sana.

1.1 Introducción

El presente documento tiene como objetivo detallar el proceso de creación de la aplicación “Fantasy Walker”, para esto, se planteará una problemática social la cual, como explicaremos más adelante en este documento, se agravo durante el periodo de pandemia sufrido debido a la crisis sanitaria causada por el COVID-19 y junto a esto se propondrá una solución que busque ayudar a adultos jóvenes a prevenir el sedentarismo teniendo como base una aplicación móvil que haciendo uso de temáticas de fantasía medieval y mecanismos de gamificación incentive el ejercicio diario.

A continuación, con el fin de lograr una buena comprensión del contenido del presente proyecto de grado, se realizará una contemplación de los conceptos que sirvan de pilares para la elaboración del documento y le permitirán a un lector inexperto en esta área poder entender sin mayor problema. Seguido de esto se verán aplicaciones y juegos de terceros que tengan relación con la problemática que estamos buscando tratar y como nuestra solución se diferencia de ellas tanto en el objetivo como en la forma de afrontar la problemática.

Una vez terminado este marco teórico, buscaremos realizar un análisis del problema donde lo que se buscara será identificar todos los requerimientos y proyecciones que pueda tener el proyecto y el sistema, esto con el fin de poder desarrollar una arquitectura haciendo uso del modelo 4 + 1 y así proceder a una descripción de las metodologías a utilizar y por último presentar como se dio el desarrollo de la aplicación y los resultados que se obtuvieron

2 Descripción general

2.1 Antecedentes, problema y solución propuesta

2.1.1 Descripción de la problemática u oportunidad

Según el ministerio de salud de Colombia el 37.7% de los adultos de 18 a 64 años tienen sobrepeso y un 18.7% sufre de obesidad para un total del 56.4% de adultos en este rango que tienen exceso de peso. El exceso de peso en adolescentes ha visto un incremento significativo en donde en 2010 el 15.5% de esta población tenía problemas de sobrepeso y en 2015 este porcentaje incremento a 17.9%. Los comportamientos sedentarios han aportado al incremento de los factores de riesgo para la obesidad, estos causados en parte debido a la reducción del trabajo físico debido a adelantos tecnológicos y reducción del tiempo dedicado al juego al aire libre (Obesidad, un factor de riesgo en el covid-19, s.f.). Estas personas están un riesgo más alto de problemas de salud que van desde condiciones debilitantes que tienen efectos adversos en la calidad de vida hasta condiciones que pueden causar la muerte prematura. La obesidad es un factor de riesgo para enfermedades no transmitidas como NIDDM, CVD y cáncer. En algunos países industrializados la obesidad también está ligada con varios problemas psicosociales. (Obesity: preventing and managing the global epidemic. Report of a WHO consultation, 2000). Ahora bien, se sabe que la falta de motivación es un factor decisivo a la hora de realizar algún tipo de actividad física, o incluso simplemente a la hora de mantener un estilo de vida sano. Esta falta de motivación puede deberse a múltiples factores entre los que se encuentran mala imagen del ejercicio, falta de compañeros con quienes ejercitarse y baja autoeficacia (Yim & Graham, 2007).

2.1.2 Formulación del problema

La falta de motivación a la hora de hacer ejercicio ha derivado en un aumento en los casos de sedentarismo. Ahora bien, sabiendo que la falta de actividad física en la población es capaz de generar un deterioro en la salud que puede llegar a ser fatal, se encuentra imperativo la creación de una motivación que empuje a la población colombiana a realizar actividad física, pero más importante aún a disfrutar realizándola.

2.1.3 Propuesta de solución

Como respuesta a la problemática se busca crear una aplicación que ofrezca una aventura como experiencia para motivar a la población de jóvenes adultos colombianos a salir de su casa y realizar actividad física. Además, se plantea promover diferentes tipos de ejercicios mediante misiones sorpresa a lo largo del trayecto para brindar una experiencia más completa, pero sin desvincular el ejercicio de la experiencia que se desea brindar. También, se plantea introducir una barra de experiencia, la cual au-

mentará de nivel a medida que el usuario haga uso de la aplicación. Esta barra estará fuertemente ligada a las recompensas dentro del juego, esto con el fin de hacer que el usuario tenga una sensación de progreso en su rutina de ejercicio.

2.1.4 Justificación de la solución

Durante los últimos años se han visto varios ejemplos de aplicaciones o juegos de consola que promueven *gamificar* en parte el ejercicio. Pokémon Go es probablemente el ejemplo más conocido, ya que durante los primeros meses de su lanzamiento fue extremadamente popular. A pesar de que el juego no estaba enfocado en ser una aplicación para ejercicio, el juego logro motivar a cientos de miles de personas a salir y caminar, en muchos casos grandes distancias al día, para buscar “capturar” las criaturas virtuales conocidas como Pokémon. Otro ejemplo también muy exitoso es el de Ring Fit Adventure, también producido por Nintendo, que busca transformar el ejercicio en mecánicas del juego. Se pueden encontrar en internet muchas anécdotas de personas que indican haber tenido una vida muy sedentaria, sobre todo debido a las restricciones impuestas por el COVID 19, y que Ring Fit Adventure los motivo a por fin dejar su rutina sin actividad física ya que Ring Fit Adventure les provee de entrenamiento al mismo tiempo que hacen ejercicio. Estos ejemplos demuestran que la gamificación del ejercicio es exitosa para motivar a las personas a realizar actividad física ya que reducen la monotonía y los aspectos “aburridos” del ejercicio.

2.2 Objetivos

2.2.1 Objetivo general

Desarrollar una aplicación móvil para el sistema operativo Android cuyo objetivo es motivar a la población Bogotana de adultos jóvenes a salir de su casa y realizar actividad física.

2.2.2 Objetivos Específicos

- Diseñar contenido multimedia, que acompañe la temática de fantasía que incremente el atractivo de la aplicación.
- Diseñar eventos narrativos modulares que serán usados para crear una historia que *gamifique* el ejercicio.
- Implementar algoritmos que permitan la generación de rutas semi - aleatorias con origen igual al destino basado en personalización del usuario.
- Validar realización de ejercicios planteados, mediante el uso de sensores del dispositivo móvil mientras este se encuentre en el bolsillo del usuario.

2.3 Entregables, estándares utilizados y justificación

En la tabla a continuación se exponen los entregables a realizar

Entregable	Estándares asociados	Justificación
Documento de conceptualización de la aplicación	IEEE	En este se va a explicar el objetivo de la aplicación, las mecánicas a <i>gamificar</i> y las mecánicas que le permitan al usuario obtener una sensación de progreso
SSD	UML, IEEE, SCRUM	Este se debe realizar previo a la elaboración del código fuente, y así tener una mejor idea de cómo realizar una debida estructuración de este y facilitar su realización
Código Fuente		Código fuente de la aplicación
Documento de pruebas	ISO-31000 Manejo de riesgos	Necesario para validar la aplicación

Tabla 1: Entregables

3 Marco teórico

3.1 Fundamentos y conceptos relevantes para el proyecto.

Con el objetivo de lograr una buena comprensión del contenido del presente proyecto de grado, se realizará a continuación una contemplación de los conceptos que servirán de pilares para la elaboración de este y le permitirán a un lector inexperto en esta área poder entender sin mayor problema.

Cabe aclarar que el presente proyecto consiste en la elaboración de una aplicación, es decir, que este se debe entender no como un videojuego más sino como una aplicación que le permita al usuario mejorar su salud, si bien se hará uso de mecánicas de videojuegos como barras de experiencias y narrativas y herramientas audiovisuales con temáticas de fantasía, el objetivo principal del proyecto a presentar es principalmente brindarle al usuario una experiencia que lo anime a realizar ejercicio y mejorar su salud física. En este contexto diremos que una aplicación es aquella que tiene funciones o usa herramientas con el propósito de hacer algo más fácil y/o realizable.

Ahora bien, para incorporar los componentes de videojuegos dentro del contexto que queremos abarcar, que es la salud física de los usuarios, haremos uso de la *gamificación*. *Gamificar* es un concepto relativamente nuevo que está atrayendo el interés de académicos y profesionales como método para mediar en el comportamiento individual. Se utiliza cada vez más en una amplia gama de contextos sociales en una variedad de actividades diversas, como la educación, el cuidado de la salud y la productividad personal (*Eoghan, Mackessy, Geary, Noonan, & Buckley, 2019*) (*Koivisto & Hamari, 2014*). En pocas palabras, la gamificación son una serie de mecanismos con los cuales se busca añadir elementos de videojuego, tales como narrativas o técnicas de diseño, a problemas o temas ajenos a los videojuegos. Así pues, se hará uso de mecanismos de gamificación que nos permitan integrar elementos de videojuegos a nuestra aplicación tales como elementos audiovisuales, por elementos audiovisuales vamos a entender todos los elementos visuales y de audio que pertenecen al componente estético de nuestra aplicación tales como los elementos estéticos del mapa, la ambientación de ligada a la temática a trabajar, los audios y las narrativas de las aventuras del usuario, entre otros.

Se plantea una temática de fantasía medieval, es decir, se harán uso de temáticas y criaturas fantásticas como hadas, dragones, monstruos, etc. Los cuales se ubicarán en el marco de la edad media para darle un toque de magia y fantasía a la experiencia que se quiere brindar en el proyecto, sumado a esto, se busca integrar elementos de juegos de rol que le permitan al usuario sentirse el aventurero en su propia historia (*Marcus, 2014*), para lo anterior debemos aclarar que un juego de rol es aquel en el que el jugador asume el rol de un personaje en una historia creada de acuerdo a un tema, en nuestro caso fantasía medieval, así buscamos que el usuario tome el rol de aventurero en su propia historia estructurada por una serie de sucesos que tendrá que

superar para poder completar su recorrido y cumplir su objetivo. Lo anterior ira acompañado de una narrativa de los hechos, esta narrativa contempla lo que el aventurero o usuario realiza en su aventura y se plantea una narración que sea acorde al tema tratado.

Con respecto a las tecnologías que se plantea usar para la implementación del proyecto tenemos conceptos como computación móvil, este es la capacidad de ciertos dispositivos tecnológicos portables para acceder a datos e información remota mediante redes de comunicación inalámbrica, es decir, desde cualquier lugar en el que se encuentren sin necesidad de cables.

Por último, queremos tratar un concepto relacionado con la gamificación del ejercicio, este concepto es conocido como *exergaming*, nombrado así por las palabras en ingles *exercise* (ejercicio) y *gaming* (jugar videojuegos), este concepto hace referencia como su nombre lo indica a realizar ejercicio mientras se juega (Saaty, Haqq, Toms, Eltahir, & McCrickard, 2021), sin embargo, si bien en el marco de este concepto se está *gamificando* el ejercicio a punto de que se vuelva divertido, este no es el punto final del *exergaming*, el *exergaming* busca usar el ejercicio como una mecánica dentro del juego, que si bien es parecido a la gamificación del ejercicio no es lo mismo, ya que para alcanzar la meta del *exergaming* se debe hacer uso de la gamificación, lo cual ya describimos previamente, para esto podemos ver que algunos juegos como Just Dance o Pokémon Go ven el ejercicio como algo que se le pudo añadir a su juego. Así pues, podemos decir que el *exergaming* junta el ejercicio y el jugar videojuegos y para esto utiliza métodos de gamificación en actividades como caminar, bailar, hacer yoga, etc. Este es uno de los conceptos más importantes que se verán en el proyecto ya que por medio del *exergaming* se visto una mejora en la salud de las personas que lo disfrutan y no solo es su salud sino también en su vida social (Koivisto & Hamari, 2014) (Saaty, Haqq, Toms, Eltahir, & McCrickard, 2021).

3.2 Análisis de alternativas de solución

3.2.1 Alternativas de solución e impacto

Para la problemática planteada en este documento se pueden encontrar en el mercado ya varias propuestas de solución con aspectos similares, a pesar de esto estas soluciones tienen diferencias que limitan su capacidad de reducir el sedentarismo de sus usuarios que la propuesta en este documento busca superar. A continuación, se presenta un análisis de algunas de las propuestas existentes.

La primera alternativa de solución es el aplicativo móvil Pokémon Go en donde esta solución busca crear una nueva experiencia de juego para la franquicia de Pokémon™ en donde el jugador puede encontrar y capturar las criaturas conocidas como Pokémon en el exterior con la ayuda de la realidad aumentada. Esta aplicación también provee otros aspectos de juego como batallar otros jugadores, mejorar las esta-

dísticas de los Pokémon y grandes batallas grupales con el objetivo de capturar un Pokémon de mayor rareza.

A pesar de que el objetivo principal de este juego no es la actividad física esta conlleva como efecto secundario que sus usuarios caminen distancias significativas para poder cumplir con los objetivos del juego (Wang, 2021). Un aspecto importante a resaltar es que debido a que Pokémon Go no se presenta como una aplicación dirigida a la realización de actividad física, sus usuarios, en su gran mayoría, buscan jugar con el único objetivo de capturar a las criaturas y subirlas de nivel sin importar que se realice o no actividad física y de hecho todas las actividades del juego se pueden realizar desde la comodidad de un automóvil u otro tipo de vehículo que permita evadir que tener que caminar hacia los eventos del juego (Broom, Lee, Lam, & Flint, 2019).

Pokémon Go se limita en que al no ser una aplicación para ejercicio esta no le interesa en realidad verificar que sus usuarios realicen actividad física cuando la usan y la actividad física que propone la aplicación no va más allá de caminar de un lugar a otro.

Otra alternativa es el video juego para la consola Nintendo Switch conocido como Ring Fit Adventure. Este juego busca *gamificar* la realización de actividad física en el hogar al presentar el ejercicio como una aventura en el estilo de un RPG (Role Playing Game) en donde un nivel del juego se compone por una historia simple, por enemigos y obstáculos que deben ser superados a través de ejercicios realizables sin tener que desplazarse, que se presentan como ataques y mecánicas de movimiento.

Al principio de cada nivel el juego tiene una pequeña sección de calentamiento y se le explica al usuario como realizar cada ejercicio que deberá realizar en el nivel de manera correcta ya que el juego se presenta como un aplicativo de ejercicio y busca que este se realice de manera correcta y segura. El juego hace uso de los controles de la consola conocidos como los JoyCons que cuentan con sensores que al usuario atárselos a ciertas partes de su cuerpo con la ayuda de bandas incluidas con el juego o al usarlos con un anillo plástico también incluido con el juego pueden verificar que el usuario realice las actividades de cada nivel (Shimizu, et al., 2021).

La limitación principal que presenta esta solución es que es para uso exclusivo frente a un televisor o cualquier otro tipo de monitor en donde se pueda visualizar el juego, lo cual causa que el usuario se quede en su hogar y no exista tampoco un aspecto social durante la realización del juego. El juego también se ve limitado en el aspecto de que tiene un precio de entrada significativo en donde el usuario debe comprar la consola y el juego.

Finalmente existe la solución de los juegos de baile. Se pueden encontrar en el mercado muchas soluciones de este tipo, inspiradas principalmente por el famoso juego Dance Dance Revolución, pero para este documento nos vamos a concentrar principalmente por el juego Just Dance del desarrollador Ubisoft Entertainment SA. Existen varias versiones de Just Dance para la mayoría de plataformas de video juegos mo-

dernas, desde versiones para IOS y Android hasta versiones para las consolas de Play Station y Xbox, pero todas se asemejan en que estas hacen uso de los sensores que provee cada plataforma en sus controles y dispositivos para medir la capacidad del usuario de realizar los movimientos de baile mostrados en pantalla. La popularidad del juego es tan grande hasta el 2020 se seguían desarrollando versiones nuevas del juego para el Nintendo Wii que ya había sido descontinuada en el 2013.

Antes de cada nivel el usuario puede escoger la canción que desea bailar y su dificultad y al final se le muestra un puntaje que se supone determina con que tanta precisión el usuario realizó el baile. El puntaje obtenido se puede agregar a una tabla de clasificación en donde se compara con otros usuarios. El juego también cuenta con un modo multijugador para su modalidad principal en donde los jugadores compiten por obtener el puntaje más alto.

Al igual que la solución anterior esta solución requiere de un televisor o monitor lo cual limita su uso a principalmente dentro del hogar del usuario, aunque esta solución si cuenta con un aspecto social, aunque este es algo limitado. El juego también se ve limitado en que tiene un precio de entrada sin importar en que plataforma se desee jugar. Otra limitación es que debido a los costos y procesos de licenciamiento de las canciones usadas por el juego el usuario tiene una selección limitada de canciones que puede bailar y es necesario el uso de micro transacciones para la adquisición de nuevos niveles.

3.2.2 Comparación de alternativas

Teniendo en cuenta las alternativas presentadas en el punto anterior se muestra la siguiente tabla que compara las soluciones con la solución propuesta en este documento.

	POKEMON GO	RING FIT ADVENTURE	JUST DANCE	NUESTRA PROPUESTA
Mínimo de ejercicio requerido	Bajo	Alto	Medio a alto	Medio
Incentivo para hacer ejercicio	Medio	Alto	Alto	Alto
Atractivo	Alto	Bajo	Alto	Alto
Actividad en el exterior	Alto	Bajo	Bajo	Alto

Aspecto social	Medio	Bajo	Bajo	Medio
Precio de entrada	No	Si	Si	No

Basado en la tabla anterior se puede ver que nuestra solución es la más acertada ya que esta no busca incentivar la realización de actividad física sino también que el usuario salga de su casa, mientras que otras alternativas como Ring Fit Adventure y Just Dance proveen la posibilidad de realizar ejercicio, pero únicamente dentro de casa. Pokémon Go mientras que si saca al usuario de su casa no requiere de una cantidad significativa de actividad física ni incentiva al usuario a realizarla. Otra ventaja que ofrece la solución propuesta en este documento es que no existe un precio de entrada real, en las otras soluciones de ejercicio el usuario generalmente requiere tener acceso a una consola de video juegos, la cual puede costar varios millones de pesos Colombianos, y acceso al juego, que en este caso son juegos de más de 200 mil pesos Colombianos en el mercado actual mientras que para nuestra solución la mayoría de usuarios ya cuentan con un dispositivo Android capaz de correr el programa y el acceso a la aplicación es gratis.

4 Análisis del problema

4.1 Requerimientos

Tras realizar el levantamiento de requisitos e investigar acerca de los hábitos de ejercicio y de cómo adaptar estos en el contexto de los videojuegos, se genera la siguiente tabla donde se contemplaran los requisitos funcionales que se consideran necesarios para ofrecer una experiencia de inmersión agradable para el usuario, junto a estos se encuentra una valoración la numérica con la cual se busca darle un atributo de prioridad al requisito con respecto a los demás, la priorización se hará en una escala de 1 a 5, donde 1 representa la máxima prioridad y 5 la mínima.

ID	Requisito	Prioridad
RF-01	Posición en tiempo real de la persona mientras realice un circuito	1
RF-02	El sistema debe de revisar si los ejercicios fueron realizados por el usuario por medio de sensores	2
RF-03	Se debe poder ver un progreso dentro de la aplicación en forma de experiencia la cual se consigue mediante la terminación de circuitos	2
RF-04	Al elegir un circuito se debe poder elegir la dificultad de este.	3
RF-05	Al elegir un circuito se debe poder modificar la ruta diseñada	3
RF-06	Al elegir un circuito se debe poder elegir los ejercicios que se realizaran en el transcurso de la ruta	3
RF-07	El programa debe mostrar instrucciones de los ejercicios a realizar durante las rutas	3
RF-08	Al finalizar un circuito se le otorga una recompensa al jugador	4
RF-09	Se debe poder crear y eliminar una cuenta dentro de la aplicación	4
RF-10	Validar el inicio de la sesión para identificarte a través de la aplicación	5
RF-11	El programa debe permitir tener amigos dentro de la aplicación	5

Pese a que se organizan los requisitos funcionales de esta forma, se es consciente de que se debe realizar en paralelo ciertas tareas para poder cumplir con el cronograma

planteado, además de la restricción temporal se debe tener en cuenta que algunas de los requisitos con menor prioridad deben estar disponibles antes de poder implementar otros. Con base en esto planteamos un diagrama donde se muestra la ruta de ejecución a seguir.

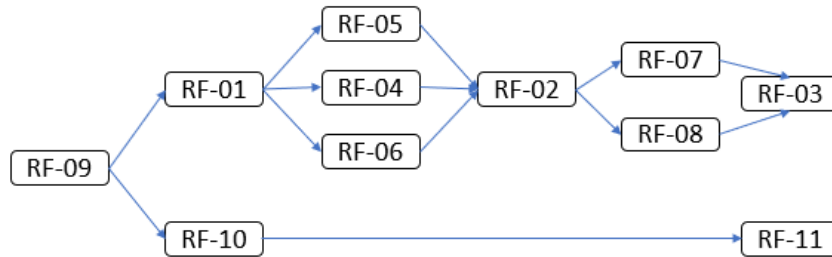


Ilustración 1 Requisitos Funcionales

ID	Requisito	Atributo de calidad
RNF-01	La aplicación tendrá un tema medieval en <i>pixel art</i>	Usabilidad: Estética
RNF-02	La aplicación debe estar en español	Usabilidad: Estética
RNF-03	No se debe demorar más de 30 segundos en iniciar sesión	Desempeño: Capacidad Temporal
RNF-04	El mapa se debe actualizar cada 0.5 segundos	Desempeño: Capacidad Temporal
RNF-05	El sistema debe saber manejar de los usuarios al no tiene activo el GPS	Fiabilidad: Tolerancia a fallos
RNF-06	La aplicación debe funcionar con 10 usuarios recurrentes	Desempeño: Capacidad
RNF-07	Información privada solo debe estar disponible para la cuenta que posee dicha	Seguridad: Confidencialidad

	información	
RNF-08	La contraseña debe cumplir con mínimo 8 caracteres y contener número y letras	Seguridad: Autenticidad

5 Análisis de impacto

Fantasy Walker se desarrolla como una aplicación móvil para dispositivos Android con el objetivo de promover la actividad física y disminuir el sedentarismo, principalmente en la población de adultos jóvenes de Bogotá. Esto se debe a que este grupo de personas es principalmente afectado por el sedentarismo debido a las largas horas que deben dedicar a su estudio y/o trabajo y recientemente debido a la pandemia causada por el virus SARS-CoV-2, donde esta población se vio aún más afectada debido a los cambios de modalidad de trabajo de presencial a virtual. A pesar de que las restricciones relacionadas a la pandemia se han en su mayoría eliminado, el cambio de hábitos durante este largo periodo todavía afecta a la población actual.

A corto plazo se espera que la aplicación tenga un número pequeño de descargas, esto debido a la falta de marketing y publicidad, y parte de esta población afectada por el sedentarismo comience a realizar algo de actividad física diaria debido al atractivo temático y de gamificación que presenta la aplicación. Posteriormente este proyecto contaría con un número significativo de usuarios y en donde Fantasy Walker sea tema de conversación y el número de usuarios incrementa principalmente debido a recomendaciones de sus usuarios. En el largo plazo el aplicativo causa una reducción no trivial del índice de sedentarismo, es una aplicación fácilmente reconocida por los adultos jóvenes de la ciudad y el gran número de usuarios permite y justifica la expansión de contenido existente e implementación de contenido nuevo para así seguir creciendo.

6 Descripción Global

6.1 Perspectiva del producto

Historia de usuario	
Número: 1	Usuario: Usuario
Nombre de la historia: Ver a mi personaje en los mapas con seguimiento en tiempo real	
Prioridad en negocio: alta	Riesgo en desarrollo: bajo
Puntos estimados: 4	Iteración asignada: 2
Programador responsable: Nicolas D. Vargas, Luis Peñaranda	
Descripción: Como usuario, quiero poder ubicarme en un mapa en tiempo real teniendo como referencia a mi personaje dentro del juego	
Criterios de aceptación: El mapa muestra la posición del usuario en el mapa y se muestra como marcador al personaje del usuario.	
Validación: <ul style="list-style-type: none">-El usuario es capaz de ver a su personaje-El Usuario es capaz de ubicarse en el mapa-La ubicación del usuario en el mapa es coherente con su ubicación actual-El personaje que se muestra en pantalla es el usuario del personaje	

Tabla 1 historia de usuario 1

Historia de usuario	
Número: 2	Usuario: Usuario
Nombre de la historia: Creación y personalización del circuito	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 32	Iteración asignada: 2
Programador responsable: Nicolas D. Vargas, Juan Camilo Merchán	
Descripción: Como usuario quiero poder crear y personalizar unas rutas por las cuales realizar mi ejercicio.	
Criterios de aceptación: Se logra crear exitosamente una ruta con las especificaciones dadas por el usuario	

<p>Validación:</p> <ul style="list-style-type: none"> -Él se crea exitosamente una ruta entre un punto A y un punto B -La ruta es coherente con la ubicación del usuario y la distancia a recorrer -El usuario es capaz de ver la ruta en su mapa

Tabla 2 historia de usuario 2

Historia de usuario	
Número: 3	Usuario: Usuario
Nombre de la historia: Realizar el circuito	
Prioridad en negocio: Alta	Riesgo en desarrollo: bajo
Puntos estimados: 8	Iteración asignada: 2
Programador responsable: Luis Manuel Peñaranda y Daniel Castellanos	
Descripción: Como usuario quiero poder recorrer la ruta plasmada en el mapa y que una vez llegue a mi destino genere otra ruta que me lleve de vuelta al inicio	
Criterios de aceptación: El usuario recorre el circuito y una vez llega al destino se genera una ruta de regreso al inicia	
<p>Validación:</p> <ul style="list-style-type: none"> -A medida que el usuario recorre el circuito va desapareciendo la ruta dibujada en el mapa que queda detrás de la posición del usuario -Si el usuario se sale de la ruta esta se redibuja teniendo en cuenta el destino y las personalizaciones del usuario 	

Tabla 3 historia de usuario 3

Historia de usuario	
Número: 4	Usuario: Usuario
Nombre de la historia: Ejercicios apropiados para el usuario	
Prioridad en negocio: media	Riesgo en desarrollo: Medio
Puntos estimados: 8	Iteración asignada: 3
Programador responsable: Juan Camilo Merchán, Luis Manuel Peñaranda	
Descripción: Como usuario quiero que los ejercicios que se me asignan en los even-	

tos sean apropiados para mi
Criterios de aceptación: La retroalimentación dada al final del circuito por el usuario sobre cómo se sintió realizando los ejercicios es coherente con la dificultad inicial del circuito
Validación: -Se compara la dificultad inicial del circuito especificado por el usuario, con la dificultad con la que el usuario experimento los eventos a lo largo del este.

Tabla 4 historia de usuario 4

Historia de usuario	
Número: 5	Usuario: Usuario
Nombre de la historia: Validación de ejercicios por medio de sensores	
Prioridad en negocio: Alta	Riesgo en desarrollo: bajo
Puntos estimados: 64	Iteración asignada: 3
Programador responsable: Camilo Merchán y Luis Manuel Peñaranda	
Descripción: Como usuario quiero que la aplicación pueda decirme cuando y/o si realice los ejercicios dados por el evento	
Criterios de aceptación: Los sensores leen, de acuerdo con lo estipulado en el código, el movimiento del usuario para rectificar si se está o no realizando el ejercicio	
Validación: -El sensor recibe lecturas coherentes con el ejercicio realizado en el evento. -Una vez el sensor confirme la realización de la actividad permite continuar con la ruta y se le da una recompensa usuario -El sensor lee los datos de acuerdo con lo q se le indico.	

Tabla 5 historia de usuario 5

Historia de usuario	
Número:	Usuario: Usuario
Nombre de la historia: Quiero que las narraciones sean claras y coherentes con lo que se tiene que hacer	
Prioridad en negocio: alta	Riesgo en desarrollo: Medio
Puntos estimados: 32	Iteración asignada: 6

Programador responsable: Camilo Merchán y Nicolas Vargas
Descripción: Como usuario quiero que la narrativa sea coherente con el evento y que sea clara con la actividad que debo realizar
Criterios de aceptación: La narrativa del evento es coherente con lo que se muestra en pantalla y la instrucción es clara
Validación: -La narrativa coincide con el módulo de la historia correspondiente con el evento -La narrativa es clara y concisa sobre lo que debe hacer el usuario para completar el evento

Tabla 6 historia de usuario 6

6.1.1 Funciones del producto

Las funciones del producto se dividen en las cosas que el usuario podrá realizar dentro de la aplicación, las cuales se dividirán en tres

1. Mapas y circuito: El usuario podrá ver su posición en el mapa en tiempo real de donde se encuentre junto con los marcadores personalizables que este ya puso, sumado a esto, se podrá dibujar una ruta de un punto A, a un punto B y de vuelta al punto A, este circuito respetara las personalizaciones al mapa que hizo el usuario.
2. Eventos sorpresas: el usuario se encontrará a lo largo de su circuito diferentes eventos que sorpresa los cuales invitaran al usuario a realizar diferentes ejercicios, estos continuaran con la temática de la aplicación y se hará uso de los sensores disponibles en los dispositivos para validar la realización de estos ejercicios propuestos por los eventos
3. Multijugador: el usuario será capaz de ver los personajes de otros usuarios en el mapa y de igual manera será capaz jugar con ellos

6.1.2 Características de usuario

Fantasy Walker es una aplicación enfocada en un único usuario por dispositivo con pequeños aspectos sociales en línea.

Usuario registrado	Descripción
Nivel de seguridad o de privilegios	Es un usuario que en la base de datos

	<p>cuenta con un id, otros datos como nombre, correo electrónico y personalización de la aplicación, y un historial de progreso.</p> <p>Puede comenzar un circuito en donde al finalizarlo se le da una puntuación por haber realizado las actividades en el circuito y esto se guarda en la base de datos.</p> <p>El usuario puede editar su perfil, incluyendo la eliminación de su cuenta.</p>
Roles	Jugador: El jugador tiene acceso a la creación de sus circuitos y su personalización, así como su eliminación.
Nivel de estudios o experiencia técnica	<p>Este usuario debe tener la capacidad de realizar las funciones básicas relacionadas al sistema operativo Android, como instalar y correr una aplicación.</p> <p>El usuario debe tener la capacidad de leer y escribir.</p> <p>El usuario debe tener una capacidad básica de medir su capacidad física para realizar la personalización de circuitos adecuadamente.</p>
Frecuencia de uso	Se espera que este usuario haga uso diario de la aplicación en donde completa un circuito cada día.

Tabla 7 características del usuario registrado

Usuario no registrado	Descripción
Nivel de seguridad o de privilegios	Es un usuario que no se encuentra en la base de datos y solo tiene acceso a la funcionalidad de registrarse.

Roles	NA
Nivel de estudios o experiencia técnica	<p>Este usuario debe tener la capacidad de realizar las funciones básicas relacionadas al sistema operativo Android, como instalar y correr una aplicación.</p> <p>El usuario debe tener la capacidad de leer y escribir.</p>
Frecuencia de uso	Este usuario hace un uso único de la aplicación y se espera que pase a ser un usuario registrado.

Tabla 8 características del usuario no registrado

Administrador	Descripción
Nivel de seguridad o de privilegios	<p>Es un usuario que en la base de datos cuenta con un id, otros datos como nombre, correo electrónico, etc.</p> <p>Tiene acceso casi total al <i>backend</i> en donde puede editar los datos públicos de los usuarios registrados, así como crear y eliminar usuarios registrados.</p> <p>El usuario puede editar su perfil, incluyendo la eliminación de su cuenta</p>
Roles	Administrado: Cuenta con las herramientas para administrar el <i>backend</i> de la aplicación.
Nivel de estudios o experiencia técnica	<p>Este usuario debe tener la capacidad de realizar las funciones básicas relacionadas al sistema operativo Android, como instalar y correr una aplicación.</p> <p>El usuario deber tener la capacidad de</p>

	administrar <i>backends</i> desde <i>endpoints</i> HTTP, así como interactuando directamente con él.
Frecuencia de uso	Se espera que este usuario acceda al <i>backend</i> de forma diaria para solucionar problemas con este.

Tabla 9 características del administrador

6.2 Supuestos y restricciones del producto

Los supuestos y restricciones del producto incluyen toda situación o elementos que afecten el desarrollo y entrega de este. A continuación, se detallan los supuestos y restricciones de hardware y de software que se han identificado.

6.2.1 Supuestos

- La información de tipo privada del usuario no se compartirá con terceros ni con los administradores del juego a menos que el usuario de explícito permiso de compartirla.
- El desarrollo del proyecto tiene como fecha límite el final del semestre académico 2022-3.
- La aplicación es desarrollada para uso por usuarios que entienden el lenguaje español.

6.2.2 Restricciones de Software

El desarrollo de la aplicación se llevará a cabo en el IDE Android Studio el cual brinda las herramientas necesarias para crear y mantener aplicaciones nativas para el sistema operativo Android. El lenguaje de programación que se utilizará será Kotlin ya que es uno de los lenguajes soportados por el IDE y uno de los lenguajes oficiales para el desarrollo de aplicaciones para Android.

La versión mínima de Android para la que se desarrollara la aplicación es Android 8.0 ya que más del 85% de los usuarios de Android se encuentran en versiones equivalentes o más recientes que esta (Google Developers, 2022). También se quiere tener un número reducido de versiones a soportar para agilizar el desarrollo del proyecto.

6.2.3 Restricciones de hardware

Las restricciones de hardware se ven más que todo afectadas por la elección del sistema operativo Android como la plataforma en donde se ejecutara la aplicación.

Hardware ejecutar la aplicación: Dispositivos Android que cumplan con el requerimiento de versión planteado anteriormente. El dispositivo también debe contar con GPS y acceso a internet a través de redes de comunicación celular. Finalmente, el

dispositivo debe contar con los sensores que se van a usar para la validación de los ejercicios propuestos.

Hardware para el desarrollo:

OS: 64-bit Microsoft® Windows® 8/10/11, MacOS® 10.14 (Mojave) o superior, cualquier distribución de Linux de 64-bit que soporte Gnome, KDE, o Unity DE; GNU C Library (glibc) 2.31 o superior.

Sistema: CPU con arquitectura x86_64 con soporte para virtualización, 8 GB de RAM o más, resolución de pantalla mínima de 1280 x 800 pixeles y por lo menos 8 GB disponibles de almacenamiento (Google Developers Training team, 2022).

6.3 Requerimientos específicos

6.3.1 Interfaces con el usuario

En las interfaces de usuario se tiene en cuenta las pantallas que vera el usuario y los requerimientos de estas dependiendo de las historias de usuario. Como podemos ver se cuentan con 8 pantallas propuestas, nuestra pantalla principal será la pantalla mapa desde el cual se puede acceder a las demás pantallas relacionadas con las funcionalidades de la aplicación

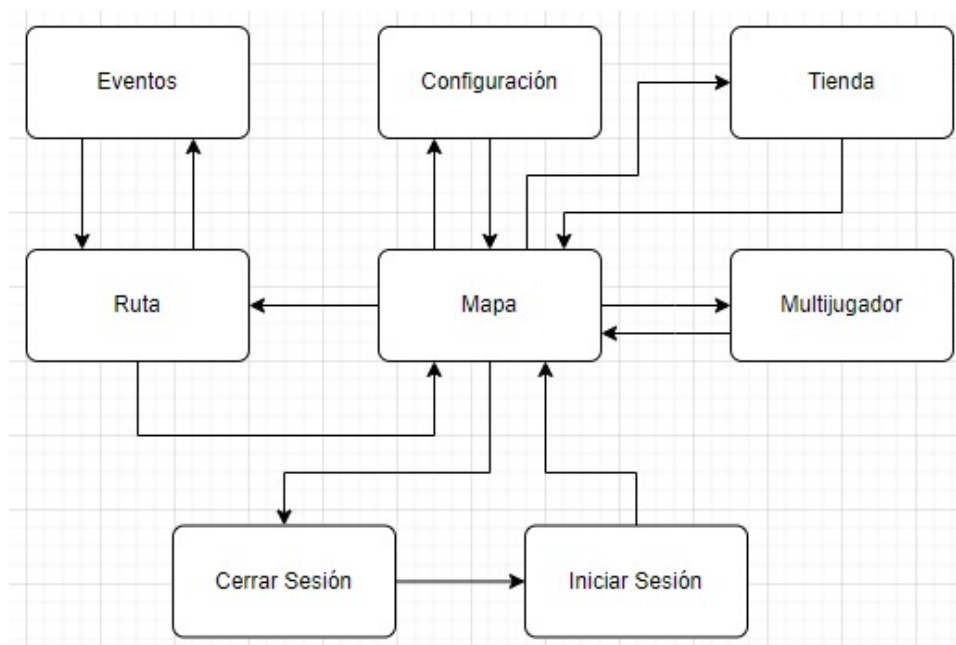


Ilustración 2 interfaces de usuario

6.3.2 Interfaces con el sistema.

En las interfaces del sistema se tiene en cuenta los diferentes componentes del sistema y como interactúan entre sí, desde la interfaz gráfica se accede a las funcionalidades de la aplicación la cual luego usa los sensores del sistema para la comprobación de los ejercicios

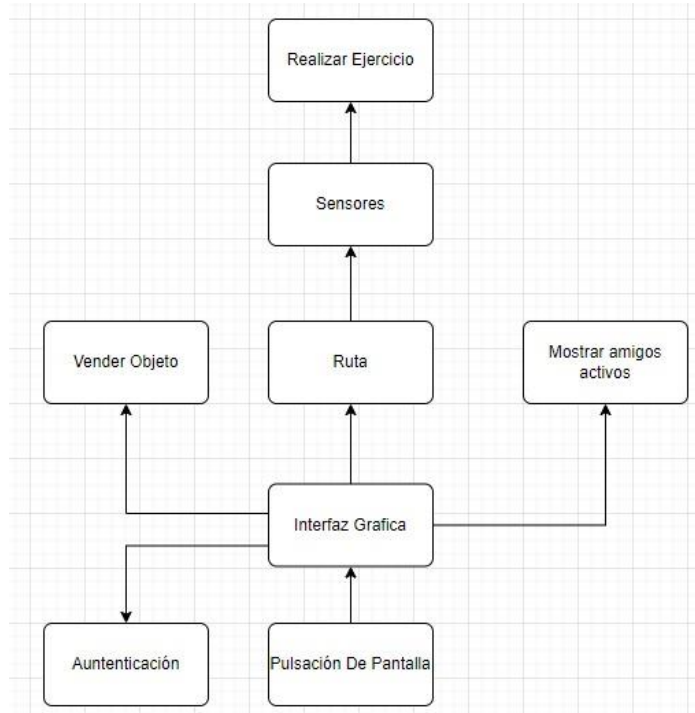


Ilustración 3 interfaces del sistema

6.3.3 Atributos de calidad

A continuación, se exponen las historias técnicas de cada uno de los atributos de calidad presentes en el desarrollo del videojuego. Cabe anotar que estos se definen dentro del estándar ISO/IEC 25000 (ISO 25000, 2022).

Tabla 10 Historia técnica - Atributo de calidad Usabilidad

<i>Atributo de calidad</i>	Usabilidad	<i>Historias de usuario asociados</i>	Todas
<i>Descripción</i>	Se construirá una aplicación que sea intuitiva para el usuario, con el fin de que este no se desvíe de la ruta principal y entiendo los eventos de manera rápida. Adicionalmente, la interfaz debe cumplir con la temática fantástica que invita a el público objetivo a usar la aplicación.		

Razón	Al ser una gamificación de hacer ejercicio, la interfaz debe ser amigable e intuitiva en el uso para el usuario final.		
Autor	Todos los integrantes del grupo.		
Criterio de medición	Revisión de la interfaz, esta debe ser intuitiva a la vista y la percepción de los eventos debe ser rápida.		
Prioridad	Alta.		

Tabla 11 Historia técnica - Atributo de calidad Funcionalidad

Atributo de calidad	Funcionalidad	Historias de usuario asociados	Todas
Descripción	Se refiere a la capacidad del aplicativo para realizar las funcionalidades para qué es desarrollado. En este caso es importante tener las mecánicas bien definidas, puesto que esta es la manera en la que el usuario va a interactuar con el aplicativo.		
Razón	Que el usuario pueda jugar en el aplicativo, y de esta manera pueda completar el entrenamiento planteado.		
Autor	Todos los integrantes del grupo.		
Criterio de medición	Revisión de una ruta planteada y como se resuelven los eventos de esta.		
Prioridad	Alta.		

Tabla 12 Historia Técnica - Atributo de calidad Disponibilidad

Atributo de calidad	Disponibilidad	Historias de usuario asociados	Todas
Descripción	Al relacionar Android Studio con Parse se garantiza que el aplicativo va a estar disponible en cualquier momento, en caso de eliminarlo y volverlo a instalar. Así como también se garantiza la disponibilidad en las copias de seguridad locales que se realicen.		
Razón	Que el usuario pueda utilizar el aplicativo en cualquier momento del día.		
Autor	Todos los integrantes del grupo.		
Criterio de medición	Ingresar en cualquier momento al aplicativo para verificar las rutas realizadas.		

Prioridad	Alta.		
------------------	-------	--	--

Tabla 13 Historia técnica - Atributo de calidad Mantenibilidad

Atributo de calidad	Mantenibilidad	Historias de usuario asociados	Todas
Descripción	Cuando se estén realizando las pruebas preliminares y se identifique un error, se debe tener la capacidad de corregirlo. Es por esta razón que el aplicativo estará construido de forma modular así un cambio en una parte no afecta todo el aplicativo		
Razón	Poder modificar errores.		
Autor	Todos los integrantes del grupo.		
Criterio de medición	Revisión por cada módulo en reuniones con todos los integrantes del equipo.		
Prioridad	Alta.		

6.4 Arquitectura 4 + 1

Tomaremos de referencia los RNF anteriormente mencionados y se compararan contra estilos de arquitectura para ayudarnos a escoger el mejor candidato para lo que queremos alcanzar como producto. Estos Requisitos no funcionales serian:

- Usabilidad-Estética
- Mantenibilidad-Reusabilidad
- Mantenibilidad-Modularidad

Teniendo en cuenta el libro de “*Software in Practice*” se hablan de 2 arquitecturas MVC (Modelo Vista Controlador) y Cliente Servidor

Cliente servidor se especifica como un estilo que quiere tratar el siguiente problema:

“Manejar una cantidad de recursos y servicios promoviendo la modificabilidad y reusabilidad del software mediante la factorización de servicios comunes y centrarlos en una sola maquina o el menor número de máquinas posibles. También se podría mediante este estilo mejorar la escalabilidad y disponibilidad. Esto se lograría mediante la centralización del control de dichos recursos y servicios que además permite distribuir los mismos en múltiples servidores físicos.”

Teniendo esto en cuenta, los puntos fuertes del estilo teniendo en cuenta los requisitos no funcionales, serian: Fiabilidad-Disponibilidad, Fiabilidad-Escalabilidad, Manteni-

bilidad-Modularidad y Mantenibilidad-Reusabilidad que son atributos que buscamos para nuestra aplicación móvil.

Sin embargo esta modularidad es solo entorno a los servicios expuestos ofrecidos en el servidor para poder lograr el requerimiento de Estética (Tanto funcionalmente para los distintos temas que pueda manejar la aplicación como para el requerimiento no funcional de Usabilidad Estética) deseada y que también sea fácil de modificar y re-utilizar en el aplicativo se propone del lado del Cliente y su interacción con el Back el modelo MVC (Modelo Vista Controlador) de modo que podremos garantizar el Requerimiento no funcional de Modularidad.

6.4.1 Descripción General del 4 + 1

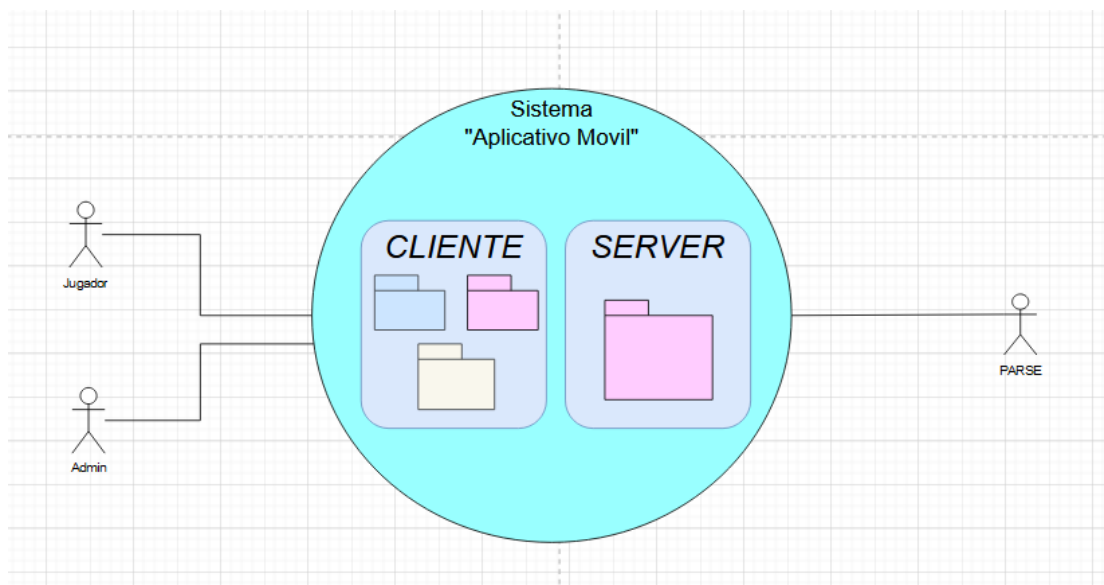


Ilustración 4 Diagrama de la vista general de la arquitectura

El sistema por diseñar es con el propósito de crear el demo del producto. Para ese Demo se priorizo el funcionamiento de la aplicación en lo correspondiente a lo requerimiento funcionales o sea el hecho de que se pueda realizar una ruta, pueda tener mi cuenta y revisar mi progreso, pueda administrar esa misma cuenta y personalizar mi menú de ejercicios entre otros especificados anteriormente. Para ello tenemos 5 actores clave en el sistema Jugador y Administrador como los usuarios del sistema, PARSE como la herramienta de base de datos y ANDROID que además de ser la plataforma sobre la cual se está desarrollando nos brinda un repertorio de herramientas como el GPS y los detectores de sensores necesarios para funcionalidades como el detectar los ejercicios.

6.4.2 Vista Lógica 4 + 1

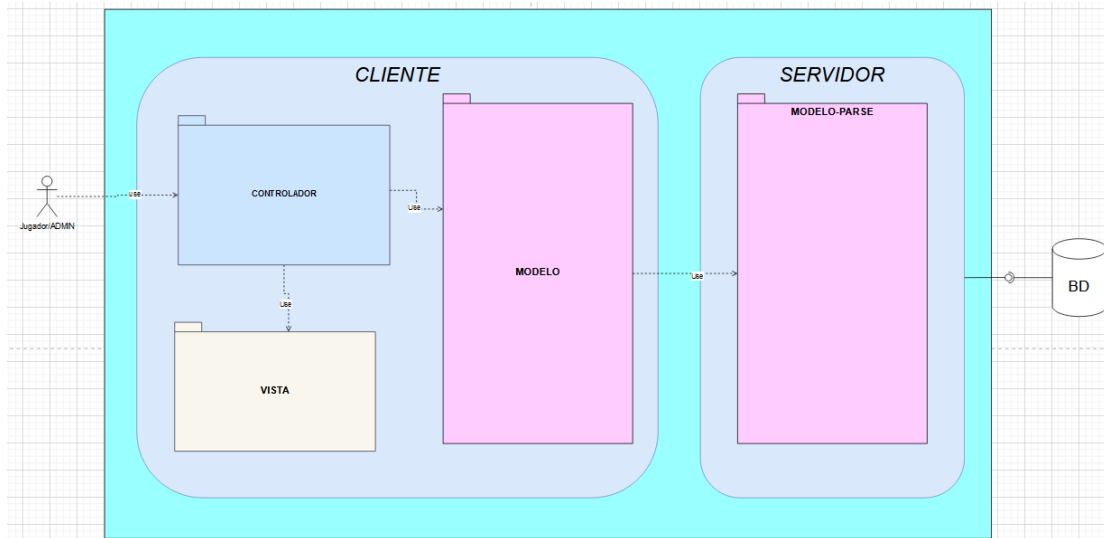


Ilustración 5 Diagrama de la Vista Lógica

Más a profundidad dentro del programa del lado del Cliente se utilizan 3 carpetas base siguiendo el estilo arquitectónico de MVC (Modelo Vista Controlador).

6.4.3 Vista de Desarrollo 4 + 1

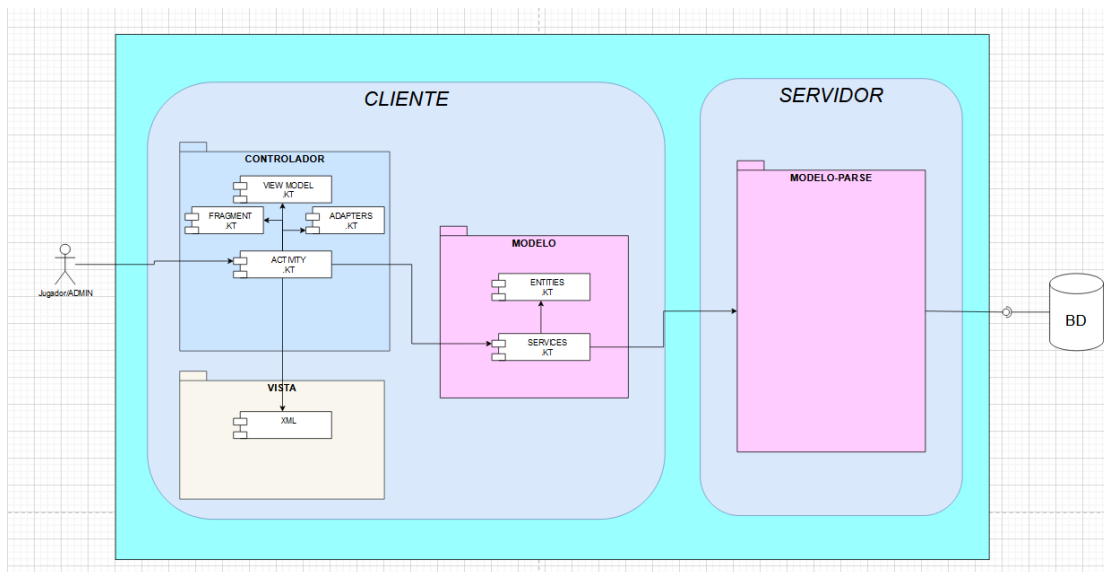


Ilustración 6 Diagrama de la Vista de Desarrollo

En el caso de Android Modelo se encargará de la estructura con la cual se entenderán los datos recibidos desde el servidor y además desde él se posibilita la comunicación con el Servidor, para el controlador se tendrán los componentes *Activity* que se encar-

garan de recibir inputs del usuario y actuar correspondiente a lo que se necesite, *Fragments* que serán componentes especializados de ciertas funcionalidades aisladas, *Adapters* para comunicación de elementos de lista y los componentes tipo *Activity* y los *View Model* para comunicación entre los elementos de un *Fragment* como botones entre otros y el *Activity* correspondiente. Finalmente, la vista que posee la estructura de las pantallas mediante archivos XML.

Del lado del servidor se tendrá la parte del modelo que se conecta a la base de datos y expone los servicios de consulta, modificación de datos de dicha base de datos.

6.4.4 Vista de Procesos 4 + 1

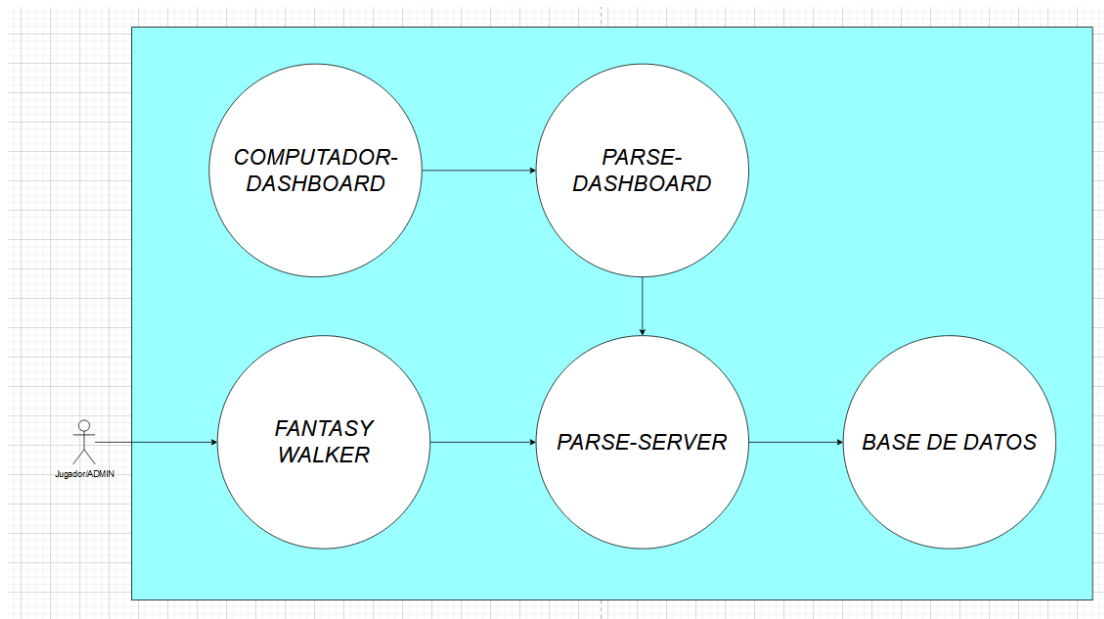


Ilustración 7 Diagrama de Procesos

En total la aplicación tiene 5 procesos importantes:

Fantasy Walker: Como el proceso que corre la aplicación móvil y la mantiene en línea.

Parse-server: Como aquel proceso que maneja la base de datos del producto y expone los servicios.

Parse-dashboard: Aplicativo Web para la administración de los datos.

Computador-dashboard: Maquina que se encargara de usar el aplicativo Web expuesto por parse-dashboard.

Base de datos: MongoDB corriendo de forma que esté disponible el mayor tiempo posible.

6.4.5 Vista Física 4 + 1

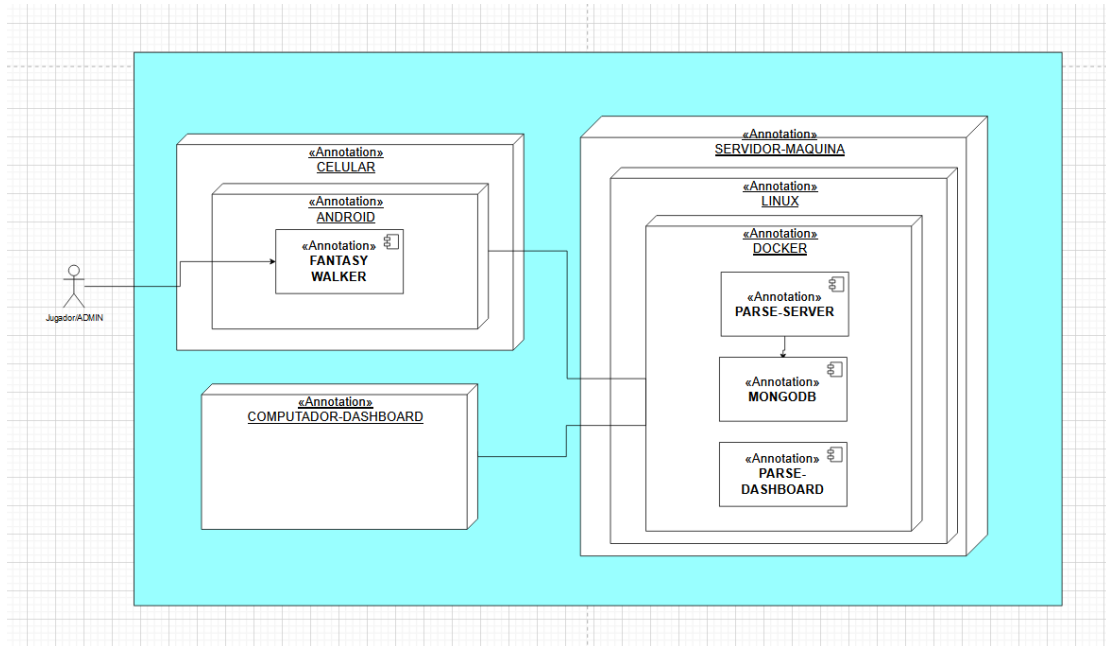


Ilustración 8 Diagrama de la Vista Física

Desde la perspectiva física se tendrían 3 tipos de máquinas importantes:

Las máquinas del celular las cuales poseerán la aplicación. Este celular debe cumplir con el requisito de que su software deberá ser Android XX.

La máquina de Servidor-Maquina es un servidor que tiene las especificaciones de poder correr parse-server, parse-dashboard y MongoDB dentro de un contenedor Docker cada uno y todo eso encerrado dentro del sistema operativo LINUX.

La máquina Computador-dashboard es una máquina con internet que correrá el cliente del Parse-Dashboard el cual es solo accesible mediante la clave correspondiente.

La comunicación manejada entre máquina Celular y máquina Servidor es mediante REST.

6.4.6 Historia de usuarios



Ilustración 9 Historias de Usuario de funcionalidad

Las historias de usuario están definidas para 3 Fases las cuales tendrán su propia organización mediante la metodología SPRINT la que se muestra para la imagen XX es correspondiente a la fase 1 del proyecto o sea la organización de la estructura de la aplicación desde lo funcional como la conexión a la base de datos, manejo de los perfiles y organización de cada pantalla.

La fase 2 se encargaría de la creación de rutas y circuitos, también el manejo de eventos y eventualmente la validación de la realización de los ejercicios. En estos casos tocaba realizar una fase de investigación sobre cómo se deben usar los sensores y de qué forma es válido revisar los ejercicios a realizar.

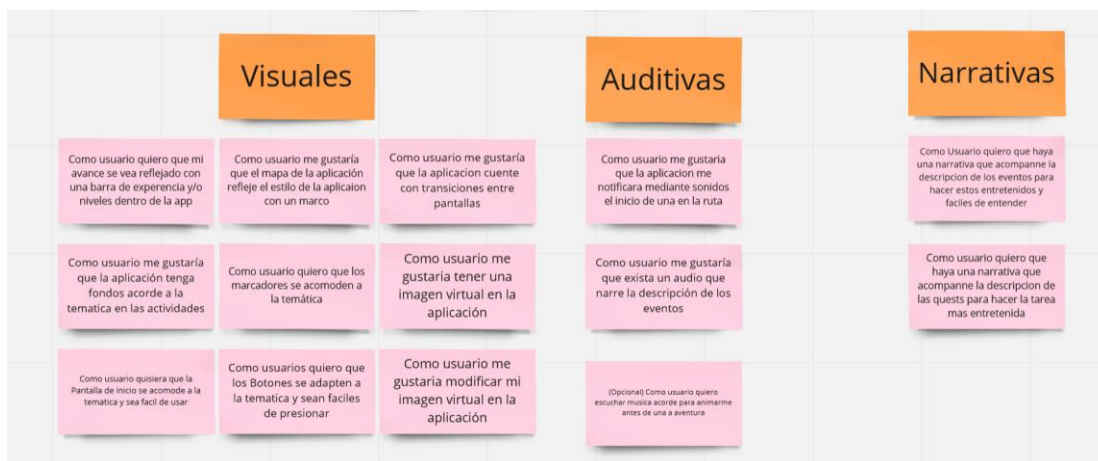


Ilustración 10 Historias de usuario de Audiovisual

Y finalmente la fase 3 que se trata de todo lo audiovisual. Esto se divide en tres categorías la visual concerniente a la forma en la que queremos se vea la aplicación lo cual corresponde a utilizar el estilo de arte Píxel Art para hacer semejanza a un video juego y la temática de aventura para crear un ambiente de aventuras de fantasía. La siguiente categoría es Auditivas la cual cuenta de sonidos ambientes dentro de la aplicación. Y finalmente las narrativas lo cual es un agregado de los eventos sobre las rutas. Esto se trata de un narrador que contara la aventura del usuario desde su inicio hasta el final de la aventura por cada aventura que el realice.

Cabe destacar que, aunque el grueso de estas historias se encuentra en lo artístico no quita el hecho de que al ser un aplicativo móvil muchas de las interacciones a realizar son también de peso programable como en el caso del narrador o las animaciones visuales.

6.4.7 Escenarios 4 + 1

Las historias que priorizar para su posterior prueba son las historias de la estética y a la funcionalidad. Se escogen estas debido a que la estética probara la usabilidad y apariencia de la aplicación mientras que las historias de usuario más importantes serán las pruebas básicas de la funcionalidad en orden de probar que se cumpla todo lo que se prometió cumplir para la versión de prueba del producto.

Estética

Las historias de usuario escogidas para juzgar la estética son:

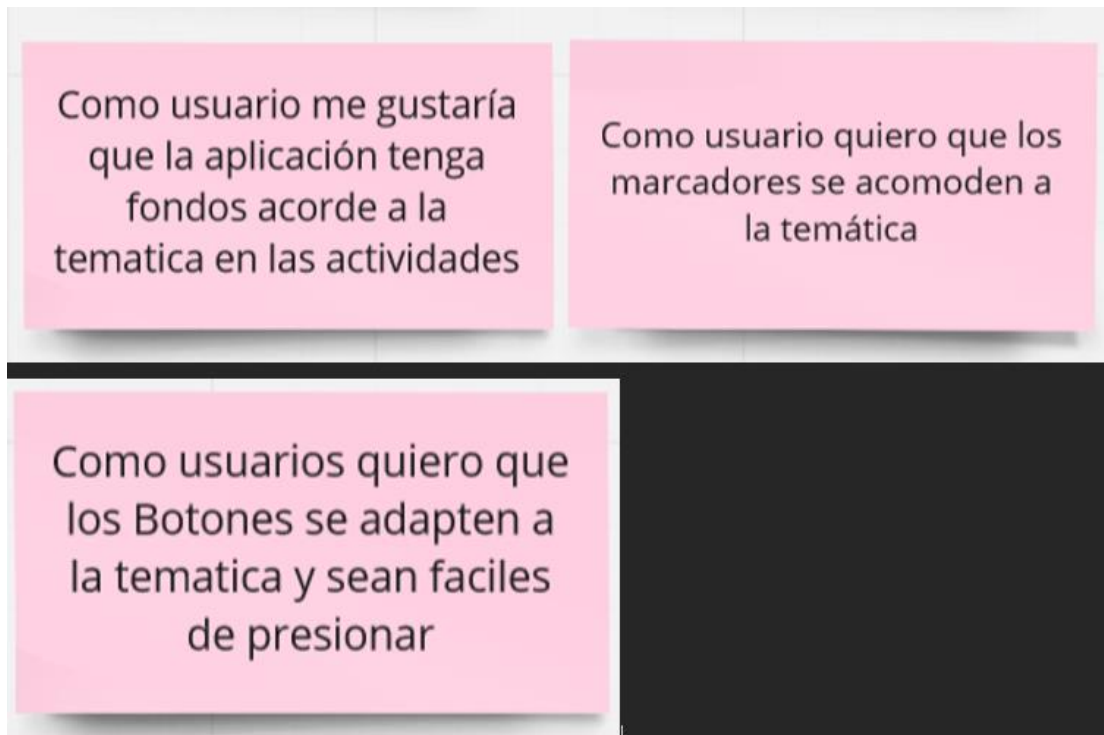


Ilustración 11 Historias importantes de Audiovisuales

Con estas historias se buscará evaluar que tan atractivo e intuitivo se ven los 2 escenarios de la parte estética de la aplicación. El escenario 1 (Sobre la historia sobre los fondos con temática medieval) es sobre ¿Qué tan atractivo se ve para ellos la temática trabajada en el formato de Píxel Art? Y que tan importante es para ellos ese visual para la inmersión de la aplicación. Y el escenario 2 (Sobre las historias de usuario de las temáticas de los componentes de interacción de la aplicación) que es sobre marcadores, botones y componentes con los que se pueden interactuar en la aplicación el objetivo es determinar qué tan intuitivo son las pantallas de la aplicación.

Funcionalidad

Todas las funcionalidades son organizadas mediante los *sprints* desde el 1 al 5 las funcionalidades destacadas se probarán bajo la idea de probar la experiencia de usuario. Estas funcionalidades destacadas son:

- Iniciar Sesión
- Crear Usuario
- Editar mis datos
- Ver mi ruta en el mapa
- Agregar/Quitar ejercicios de mi ruta
- Realizar ejercicios en mitad de la ruta
- Avance de niveles dentro de la aplicación

7 Proceso

Se hará uso de la metodología de SCRUM para el desarrollo del proyecto correspondiente. La razón de porque usaremos SCRUM es debido a la naturaleza flexible y ágil que maneja, lo cual es perfecto para nosotros por si se ve la necesidad de cambiar parte del alcance durante el desarrollo. En resumen, priorizaremos funcionalidad y el máximo contenido alcanzable con respecto al contenido audiovisual. En ese orden de ideas los temas a resolver en orden serian:

- Definición de rutas
- Definición de las pantallas de la aplicación
- Definición de la persistencia de la aplicación
- Definición de los ejercicios a realizar durante la ruta
- Definición de los elementos de gamificación

7.1 Fase metodológica 1 (Diseño e implementación de elementos básicos de la aplicación)

Esta fase se piensa diseñar e implementar el recorrido de las rutas, que tipo de rutas y los criterios de selección de eventos durante esas rutas. Con respecto a la aplicación se planea diseñar las pantallas de la aplicación e implementar el mínimo funcional como:

- *Log In*
- Persistencia
- Pantalla de la ruta
- Pantalla multijugador
- *EndPoints*

7.1.1 Método

Para adaptar todo el desarrollo a la metodología SCRUM se tiene que separar el desarrollo de la aplicación en elementos que representan la aplicación (log in, pantallas de la aplicación, etc.), elementos de funcionamiento de las rutas y el ejercicio y finalmente los elementos audiovisuales y de gamificación. Es necesario separarlo para definir el alcance final del proyecto y plantea realizar para el final del proyecto en orden de terminar el producto mínimo viable de la aplicación.

7.1.2 Actividades

En esta fase de diseño se busca tener todas las pantallas y definición de *endpoints*, las rutas (en las cuales se contempla evitar las calles por donde el usuario no quiera pasar), dificultad de las rutas planteadas y la implementación de estos hitos sin elementos estéticos.

7.1.3 Resultados esperados

Una estructura funcional de la aplicación que tenga la capacidad recorrer todas las pantallas, no posee ningún elemento estético solo las funcionalidades CRUD, la generación y utilización de las rutas, la gestión de esas rutas o sea su modificación y flexibilidad de punto inicial de ruta.

7.2 Fase metodológica 2 (Diseño e implementación de funcionalidad de ejercicio)

Esta fase corresponde a la combinación de las rutas con posibles ejercicios durante esa ruta, junto con su validación, o sea el uso de sensores para brindar una detección de la realización de los ejercicios, cabe destacar que solo se busca identificar cuantas repeticiones se realizaron no si el ejercicio realizado fue hecho de forma correcta.

7.2.1 Método

En este caso la idea es realizar el criterio de creación de las posibles combinaciones de rutas con los ejercicios y cuantas repeticiones de cada ejercicio se pueden realizar sin que esta actividad se convierta en algo repetitivo para el usuario y generar una cantidad suficiente de contenido para el producto mínimo viable.

7.2.2 Actividades

- Definición de criterio de la generación rutas
- Definición de criterio de cuantos ejercicios y cuales se pueden juntar para el circuito
- Definir cuantas rutas únicas se pueden generar
- Implementación de esos criterios para la generación del circuito con ejercicios incluidos
- Diseño de eventos narrativos que representen la historia detrás de los ejercicios realizados

7.2.3 Resultados esperados

Funcionalidad completa del uso del mapa para el desarrollo del circuito sin importar donde inicie, ni el nivel de dificultad de las rutas. Además, de la definición de los tipos de rutas a niveles narrativos o sea cuando es un jefe, una ruta normal, o una ruta especial y que significa cada uno en cada caso.

7.3 Fase metodológica 3 (Elementos audiovisuales y de gamificación)

7.3.1 Método

En esta fase, que supondremos será aquel que consuma más de nuestro tiempo, se busca definir que herramientas se piensan añadir a la aplicación en orden de *gamificar* la experiencia en orden de implementación se buscarían implementar:

- Los elementos visuales (El píxel art del mapa, los usuarios y estética de la aplicación)
- El narrador (Las historias y la grabación de las voces)

7.3.2 Actividades

- Diseño de estilo de la aplicación
- Diseño de la escritura de la cual se hará la narración y la voz de esa narración
- Definición del alcance final de la aplicación (que es posible realizar con el tiempo restante)
- Realización del arte definido en píxel art
- La implementación de narrador junto con el circuito de rutas con ejercicios

7.3.3 Resultados esperados

Un producto mínimo viable que represente la completa funcionalidad de lo diseñado, junto con el aspecto visual de la temática realizada en píxel art. Además, de la comprobación de las pruebas que definan no solo el funcionamiento de la aplicación si no también su impacto en las personas que lo usan. Se tratará el tema de pruebas más a fondo en el documento de SPMP

8 Desarrollo de la solución

Pese a que inicialmente se planteó desarrollar nuestra solución en un semestre académico, debido a la carga académica que presentamos durante el semestre planeado y a la pérdida de un miembro del grupo, nos vimos abrumados por el tamaño del proyecto, por lo que se tuvo que replantear la duración del desarrollo de la aplicación y se realizó en un periodo de dos semestres académicos

8.1 Desarrollo y funcionamiento del *backend*

El *backend* escogido para la solución final fue parse-server por Parse Platform el cual se desplego en una máquina de Linux en un contenedor de Docker. Se escogió utilizar Docker debido a que ofrecía más facilidad de uso y despliegue. El contenedor se desplego con ciertos parámetros como el nombre de la aplicación, el nombre de la llave maestra que permite la modificación de los datos de la base de datos y la dirección IP y puerto de la base de datos en donde se almacenaran todos los datos generados y subidos a parse-server. Este *backend* permite el uso de tan solo dos proveedores de bases de datos, MongoDB y PostgreSQL, se escogió utilizar un contenedor de Docker para la base de datos de MongoDB desplegado en la misma maquina en donde se encuentra el contenedor Docker de parse-server por las mismas razones explicadas anteriormente.

Una vez desplegado parse-server se necesitaba una forma fácil de poder acceder a los datos en la plataforma para la creación de clases y posteriormente para crear nuevas entradas en estas clases que servirán como contenido para la aplicación. Se utilizo la solución de parse-dashboard la cual presenta una vista web que permite modificar los datos de parse-server de manera visual dado un id de aplicación y una llave maestra. Esta solución también permite el uso de cuentas de usuario para proteger el acceso a datos de parse-server, pero para facilitar el desarrollo de la aplicación y debido a que la aplicación desarrollada es tan solo un prototipo se decidió dejar el acceso a datos sin el uso de una cuenta de usuario. El aplicativo de parse-dashboard se desplego en un contenedor Docker, pasándole por parámetro el id de aplicación y la llave maestra utilizadas anteriormente, en la misma máquina que parse-server y utilizando el proxy inverso Nginx se procedió a exponer a ambos servicios para su fácil acceso desde cualquier maquina remota a través de un dominio web y con encriptación SSL a través de HTTPS.

La máquina usada para el despliegue de esta solución fue el servidor personal de uno de los integrantes del grupo. Se utilizo esta máquina ya que, a pesar de que no se podía garantizar disponibilidad total, la disponibilidad ofrecida era lo suficientemente buena para el prototipo desarrollado durante este proyecto y que se podía dar uso ilimitado de esta máquina sin ningún costo.

Parse automáticamente crea ciertas clases, incluida la clase “*User*” en donde se guarda la información de usuarios registrados. Inicialmente se crearon algunos usuarios de prueba para facilitar el desarrollo de la aplicación móvil. Se crearon también las clases Ejercicio, Evento, *Quests* y subclases que contendrán el contenido de la aplicación, por ejemplo: en la clase ejercicio se guarda la información de los ejercicios que podrán ser asignados a un usuario durante la realización de una ruta y hace uso de la clase ImagenEjercicio para el almacenamiento de imágenes relacionadas al ejercicio con descripciones y otra información pertinente. Durante el desarrollo del proyecto se agregó más información a cada una de estas clases para expandir el contenido de la aplicación y para la realización de pruebas.

8.2 Desarrollo de la aplicación móvil

El desarrollo de la aplicación fue realizado en Android Studio utilizando el lenguaje de programación Kotlin y la comunicación con el *backend* se realizó a través del SDK que proporciona Parse Platform para Android Studio.

Inicialmente se configuro el proyecto de Android Studio para permitir su conexión con parse-server a través del uso del SDK proporcionado por Parse Platform con el id de la aplicación en parse-server y el URL en donde se encuentra el *endpoint* al *backend*. Comprobada la conexión entre la aplicación y el *backend* se crearon las clases de datos que representan la forma en la que se va a guardar la información almacenada en parse-server para su uso local. Una vez creadas estas clases se desarrollaron clases que fueron denominadas como “servicios”, que no deben ser confundidas con servicios de Android. Estas clases son las encargadas de traer la información de parse-server, transformar esta información en objetos de las clases para el uso local, lógica de negocio que se espera usar en más de una actividad, modificación y actualización de los datos. El uso de las clases “servicio” también facilita la comunicación entre actividades. Dentro de cada servicio se creó una función de inicialización en donde se trae toda la información necesaria de parse-server en el momento en el que inicia la aplicación. Durante toda la duración del desarrollo del proyecto se implementaron nuevas funcionalidades en cada servicio, pero en primera instancia tan solo se implementaron las funcionalidades de traída de información de parse-server, transformación a objetos locales y la actualización de los datos en el *backend*.

Ya implementadas las funcionalidades básicas de los “servicios” se procedió a la creación de las actividades de la aplicación por funcionalidad, con un enfoque en la funcionalidad de cada actividad y no uno visual. Primero se crearon las actividades de inicio de sesión y registro, después las actividades relacionadas con la modificación de datos del usuario, actividades de rutas, información y personalización de la aplicación.

Durante el desarrollo de las actividades relacionadas a mapas y seguimiento en tiempo real se presentaron la mayoría de los inconvenientes durante el desarrollo, ya que

se le dificultaba al equipo realizar pruebas de funcionamiento, debido a la naturaleza de la aplicación, la cual requiere salir de casa y recorrer una distancia significativa. El emulador de Android para Android Studio brinda una solución para este inconveniente permitiendo simular la ubicación del teléfono emulado e incluso simular el movimiento de este, pero debido a errores en esta simulación, como que el emulador tarda un poco en actualizar su ubicación inicial predeterminada por Android a la ubicación configurada, se presentaban errores que no aparecen durante el uso de la aplicación en un dispositivo real lo cual requirió realizar las pruebas de correcta funcionalidad dando uso del emulador y un dispositivo real constantemente.

A pesar de que se planeaba realizar todo el desarrollo funcional de la aplicación antes de la creación de contenido y el embellecimiento de las pantallas, debido a problemas en el desarrollo a causa de inexperiencia y errores, en algunas funcionalidades se terminaron de implementar después de iniciada la tarea de creación de contenido. Con cada imagen que se creaba se procedía a su importación dentro de la aplicación para así que el equipo la aprobara o se discutieran cambios a realizar.

8.3 Creación de contenido y embellecimiento de las pantallas

Durante un periodo de más o menos dos semanas, el equipo se dedicó a la creación de imágenes para el uso dentro de la aplicación con la ayuda del editor de imágenes para arte de píxel Aseprite. Las imágenes creadas incluyen fondos de botones, fondos de pantalla, fondos para bloques de texto, imágenes para el uso en eventos y *quests*, iconos, etc. Las imágenes reutilizables, como las hechas para botones, se crearon primero como una base que podía ser utilizada para la posterior creación de imágenes más detalladas. Con cada imagen creada se procede a importar la imagen a la aplicación móvil para ser aprobada por el equipo o para la realización de sugerencias de cambios a realizar. Ya terminadas las imágenes a usar se procedió a embellecer las pantallas existentes en la aplicación con el uso de estas imágenes y la reorganización de elementos en pantalla.

Posteriormente el equipo se dedicó a crear contenido en la plataforma de parse-server a través de parse-dashboard, creando entradas en las clases de *Quest* y Evento en donde cada entrada cuenta con una descripción narrativa y con imágenes que acompañan a esta descripción.

8.4 Tareas y Manejo de ramas y *commits*

Como se tenía previsto, se hizo uso de la aplicación de GitKraken para facilitar el manejo y control de ramas y *commits* a lo largo del proyecto, para hacer un buen control de errores cada integrante tenía una rama sobre la cual trabajar, esta rama llevaba el nombre del integrante dueño de esta y el sprint en el que se encontraba, esto con el fin de tener claro de donde venían los cambios y poder, en caso de ser necesario, hacer un *traceback* a una versión sin errores donde se pudiera identificar la rama causante del error o errores causantes del daño, una vez terminado el sprint, se realizaba

un *merge* de las ramas de los integrantes del equipo y posteriormente se probaban las funciones implementadas pero esta vez sobre el *main* de la aplicación, en etapas más tardías del desarrollo en lugar de nombrar la rama con el número del *sprint*, se le daba el nombre de la función específica que se estuviera trabajando en esta. Debido a que el desarrollo se alargó más de lo esperado, los nombres de algunas ramas se repitieron, por ejemplo, en el caso del primer y segundo semestre académico se repiten, ya que se tomó cada semestre como etapas diferentes del desarrollo y en cada etapa se tuvieron que plantear *sprints* nuevos. Por último, en la descripción de los *commits* se escribe brevemente lo que se realizó y GitKraken muestra los archivos que se modificaron y en donde se modificaron.

9 Resultados

9.1 Aplicación

9.1.1 Implementación

A pesar de los inconvenientes que surgieron durante el desarrollo de la aplicación se logró alcanzar varios de los objetivos planteados, tanto en el apartado gráfico como los objetivos de implementación. Durante el primer semestre cursado se logró implementar el esqueleto de la aplicación, cosas como la base de datos, los usuarios, el menú de inicio, las misiones, y el recorrido, fueron implementadas con el fin de poder centrarnos en los objetivos que se plantearon, no obstante, esto representó más tiempo de lo que se esperaba lo que repercutió en un retraso de la fecha de entrega. Lo que inicialmente se planteó entregar durante el primer semestre cursado se tuvo que retrasar y si bien no se tenía un objetivo temporal si disminuyó la moral del equipo, no obstante, esto le permitió al equipo poder realizar un mejor trabajo en los objetivos propuestos del trabajo.

Durante el segundo semestre cursado ya teniendo el esqueleto de la aplicación se buscó alcanzar los objetivos especificados, así pues, lo primero que se buscó alcanzar fue lo relacionado al código, es decir, la implementación de las rutas semi – aleatorias y la validación de ejercicios mediante el uso de sensores. Para las rutas semi – aleatorias se decidió permitirle al usuario elegir los puntos por los que quería pasar, de esta forma la aplicación no recomendará lugares peligrosos y el usuario puede elegir si hay algún destino, como un parque o tienda, por el cual desee pasar, no obstante, la ruta intermedia es una ruta generada por OSMdroid, esta será la ruta más rápida y en cierta forma depende del mapa donde se esté usando la aplicación lo que agrega un poco de aleatoriedad al resultado. Si bien el objetivo se ve completo, el equipo sintió que no se cumplió en su totalidad, puesto que al existir el riesgo de que la aplicación genere rutas peligrosas para el usuario se decidió optar por algo más seguro y menos aleatorio.

Para la lectura de ejercicios usando sensores primero se implementaron lo que en los objetivos nombramos como: eventos narrativos modulares. Estos eventos están conformados por: un título, una imagen, una descripción y un ejercicio, y están divididos en tres tipos, eventos: de inicio, eventos intermedios y eventos finales. Los eventos finales están ligados con la misión inicial, mientras que los eventos iniciales y los eventos intermedios son modulares y van a cambiar en cada misión en función de los ejercicios que tenga activados el usuario.

Teniendo en cuenta que el algoritmo conoce que tantas calorías en promedio quema cada ejercicio, se calcula internamente las repeticiones que debe hacer dependiendo de la ruta y de las características físicas del usuario y esto se le añade a la descripción cuando se muestra en pantalla al usuario mientras realiza la ruta y le sale un evento.

Cuando sale el evento en pantalla se empieza a hacer uso del velocímetro para captar el movimiento del celular, y se usa un rango de valores para determinar cuándo se cuenta el movimiento como válido, esta cuenta en los ejes que mejor apoyen al ejercicio. Si bien se planteó el uso de tecnología de aprendizaje de máquina para mejorar el output de esta actividad, no se implementó debido a que esto haría que nos retrasáramos en el cronograma y perdiéramos otra cita de entrega.

Una vez implementado esto se procedió a probar, el resultado, como se puede ver en la ilustración 12, es un *fragment* que se superpone al mapa y permanece ahí hasta completar las repeticiones o hasta alejarse lo suficiente del lugar donde surgió el evento, esto con el fin de darle al usuario la opción de realizar o no el ejercicio planteado.



Ilustración 12 *fragment* del evento en la aplicación

Por último, con el fin de desarrollar contenido multimedia que acompañe la temática de fantasía de la aplicación, se hizo uso de la aplicación de Aseprite para crear botones, fondos, indicadores y demás elementos en el estilo de *pixel art*. En el caso de los botones se hicieron dos para cada opción, uno normal con forma de pergamino y otro igual pero oscurecido que simule el botón oprimido, en la ilustración 13 y la ilustración 14 podemos ver un ejemplo de esto.

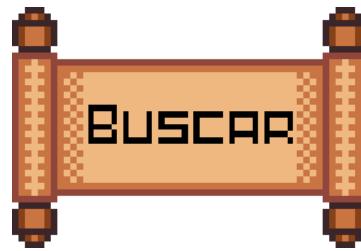


Ilustración 13 formato de botón sin oprimir



Ilustración 14 formato de botón cuando es oprimido en la aplicación

9.2 Pruebas

9.2.1 Metodología elegida

Para el caso de este proyecto en donde lo que más importa evaluar es la experiencia del usuario frente a la aplicación, realizamos cuestionarios teniendo como base la metodología de QUIS (*Questionnaire for User Interaction Satisfaction*). Esta metodología está diseñada para poder medir la satisfacción subjetiva de los usuarios con la interfaz de un sistema computacional. QUIS contiene varios cuestionarios que incluyen un cuestionario demográfico, un cuestionario de satisfacción general con la aplicación y cuestionarios de ciertos aspectos específicos como terminología, retroalimentación, factores de aprendizaje, etc. (Harper & Norman, 1993). Cabe aclarar que, no se siguió al pie de la letra la metodología QUIS, sino que meramente nos basamos en su método de evaluación e hicimos uso de algunas de sus preguntas.

No se escogió una metodología para pruebas de estrés ya que el *backend* de la aplicación está corriendo en parse-server, la cual es una solución abiertamente distribuida, usada para aplicación de gran tamaño a nivel global y en general aprobada por la comunidad. Debido a que la aplicación presentada no es más que un prototipo tampoco se vio necesario realizar pruebas a la solución de despliegue del *backend*.

9.2.2 Implementación de la metodología

Para implementar la metodología de QUIS se hizo uso de la herramienta de Google Forms. Se realizaron dos cuestionarios en donde cada uno cuenta con una parte demográfica que nos permite discernir la forma en la que diferentes grupos disfrutaban de la aplicación y así identificar si nuestro grupo objetivo si disfrutó de esta. Se dio uso a dos cuestionarios ya que la metodología QUIS no abarca todos los aspectos de la

aplicación que se deseaba probar, por lo que se creó un cuestionario que se apega en su mayoría a las especificaciones de QUIS y otro en el que se realizan preguntas más específicas acerca del uso de la aplicación, pero implementado de una manera inspirada por la metodología pero que deriva en sus preguntas.

El primer cuestionario contiene secciones relacionadas a demografía, satisfacción general, pantallas de la aplicación, terminología e información del sistema, aprendizaje, capacidades del sistema y usabilidad e interfaz de usuario en donde cada sección contiene preguntas con una escala lineal de 0 a 9 para su respuesta en donde 0 representa el peor valor mientras que 9 representa el valor máximo.

- Demografía: esta sección busca dar a entender las capacidades técnicas del usuario frente al uso de dispositivos móviles.
- Satisfacción general: en esta sección se busca entender cuál fue el sentimiento general del usuario hacia la aplicación.
- Pantallas de la aplicación: esta sección trata de la organización de los elementos en pantalla ya la navegación entre estas.
- Terminología e información del sistema: acá se busca responder a preguntas que tienen que ver con la consistencia y efectividad de la terminología usada en la aplicación y los mensajes informativos que muestra la aplicación para ayudar al usuario.
- Aprendizaje: en esta sección se pregunta acerca de la curva de aprendizaje de la aplicación.
- Capacidades del sistema: en esta sección se busca responder a aspectos relacionados a las capacidades computacionales de la aplicación como su velocidad y fiabilidad.
- Usabilidad e interfaz de usuario: esta sección identifica la opinión del usuario acerca de los aspectos visuales de la aplicación y su fluides.

El segundo cuestionario se centra en ver que tan intuitivo y atractivos son las pantallas presentadas por la aplicación. En términos de atributos de calidad estaríamos probando con esto los atributos de Usabilidad Estética y Usabilidad Aprendizabilidad lo cual es muy relevante para nosotros ya que este nos permitirá evidenciar el cumplimiento de hacer una aplicación atractiva y amigable de usar. Se evaluará según el QUIS o sea que la puntuación que se va a manejar es de 0 a 9. El criterio para evaluar que tan intuitivo es se basa en las siguientes preguntas (cada pregunta mostrara una imagen con la que se especifica la pantalla que se está preguntando):

- Por la organización que se maneja de los elementos en la pantalla ¿Es posible entender todo lo que se quiere hacer?

Las personas seleccionadas para la realización de pruebas fueron miembros de los círculos familiares y de amigos de los miembros del grupo. Estas personas recibieron los enlaces a ambas encuestas, instrucciones de alguno de los miembros de grupo del proyecto acerca de cómo llenar el formulario, el apk de la aplicación e instrucciones

de como instalar el apk en su dispositivo. Debido a la naturaleza del apk las personas escogidas debían contar con un dispositivo Android y debían estar dispuestas a instalar una aplicación por fuera de la Play Store, lo cual la mayoría no habían realizado antes y algo de lo que personas desconocidas desconfiarían.

9.2.3 Resultados de pruebas

A continuación, se muestran los gráficos relacionados a cada sección de la primera encuesta en donde cada barra representa una pregunta y el valor de la barra representa la respuesta promedio obtenida.

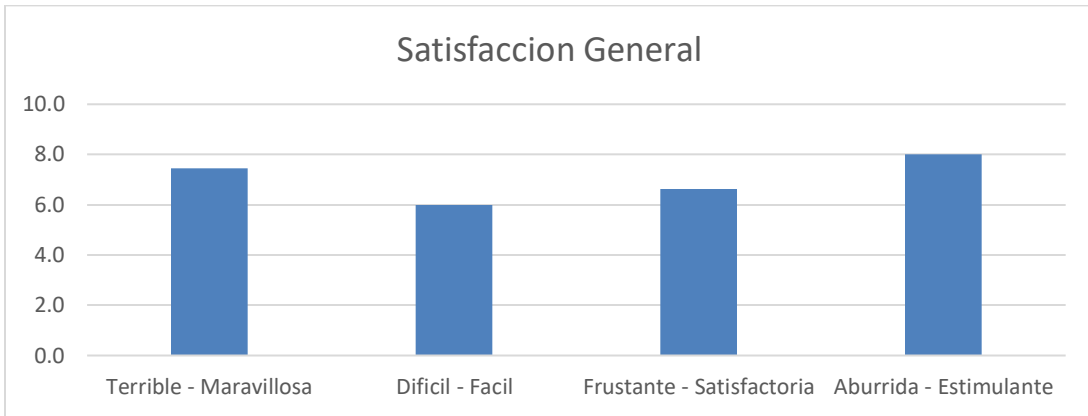


Ilustración 15 Grafico de valores promedio a las respuestas de la encuesta de satisfacción general.

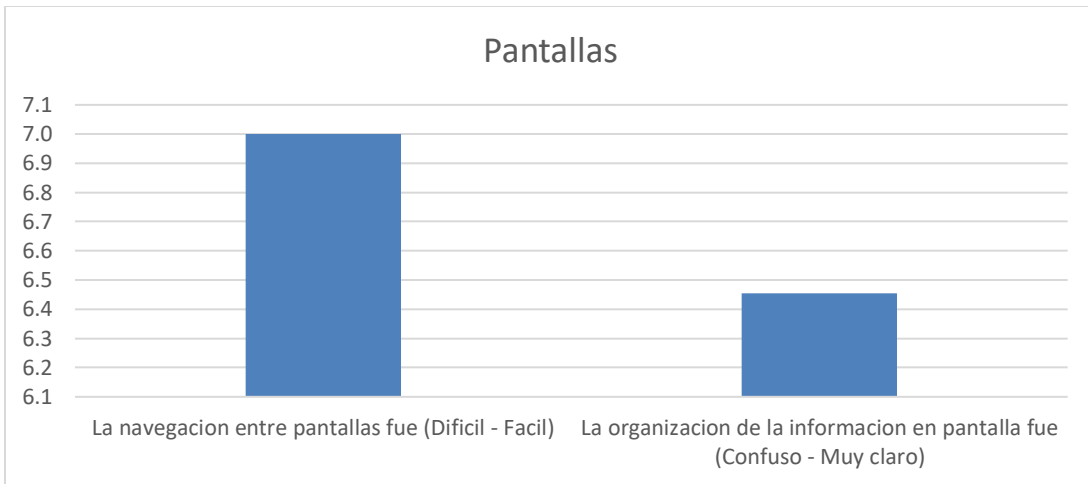


Ilustración 16 Grafico de valores promedio a las respuestas de la encuesta de pantallas.

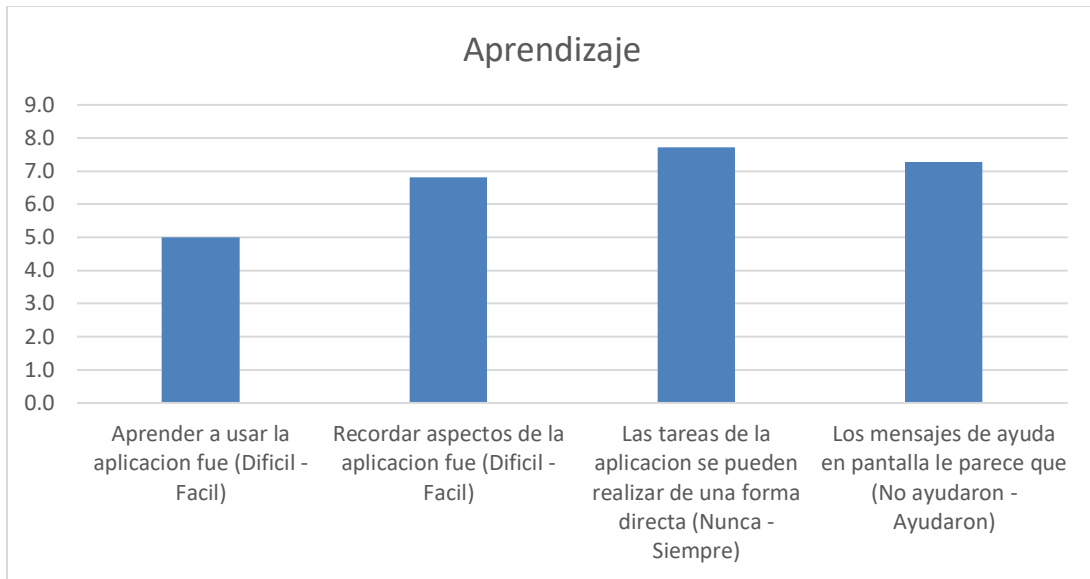


Ilustración 17 Grafico de valores promedio a las respuestas de la encuesta de aprendizaje.

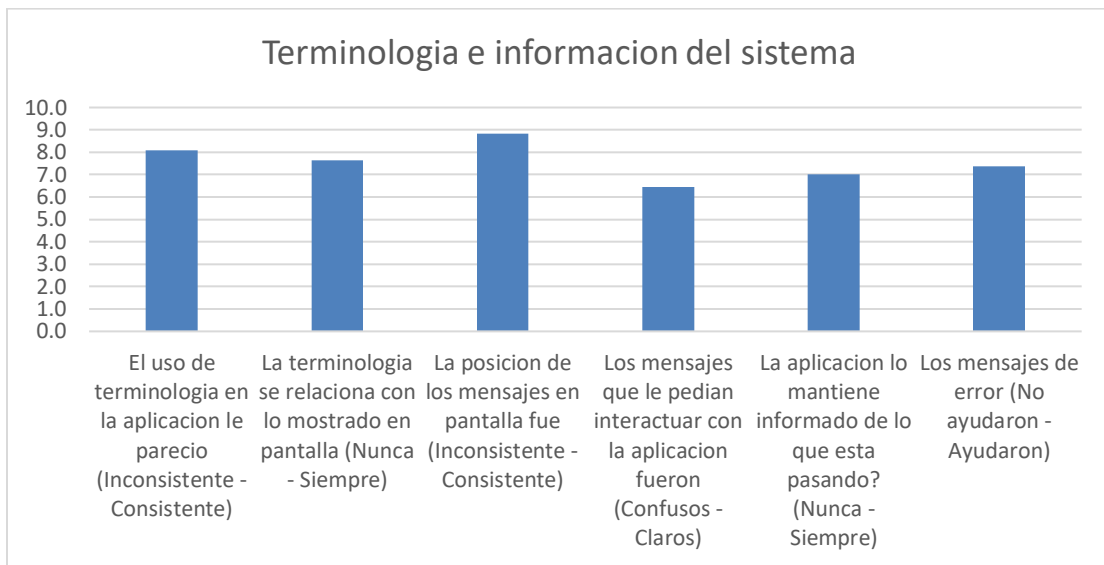


Ilustración 18 Grafico de valores promedio a las respuestas de la encuesta de terminología e información del sistema.

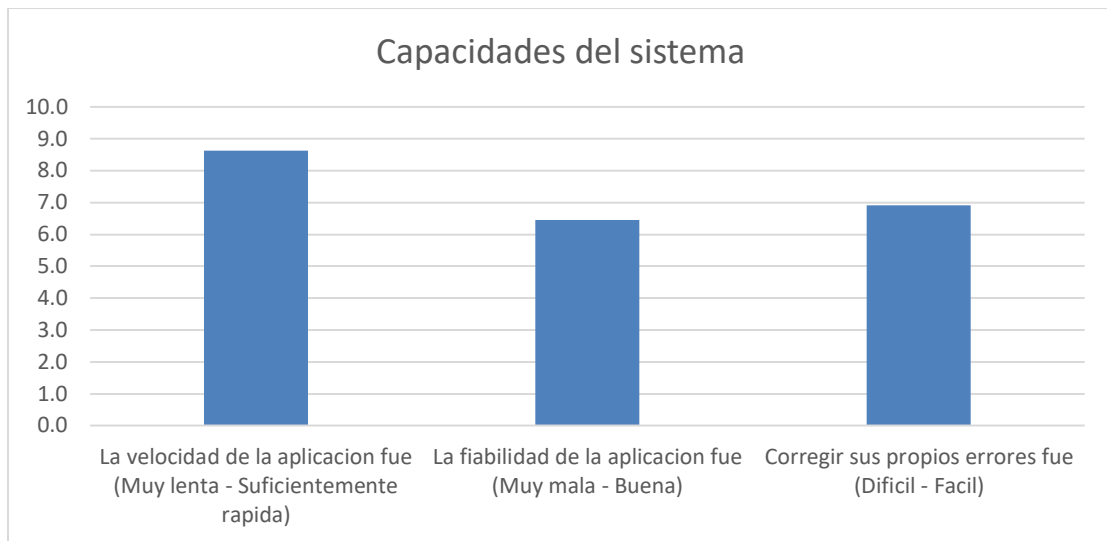


Ilustración 19 Grafico de valores promedio a las respuestas de la encuesta de capacidades del sistema.

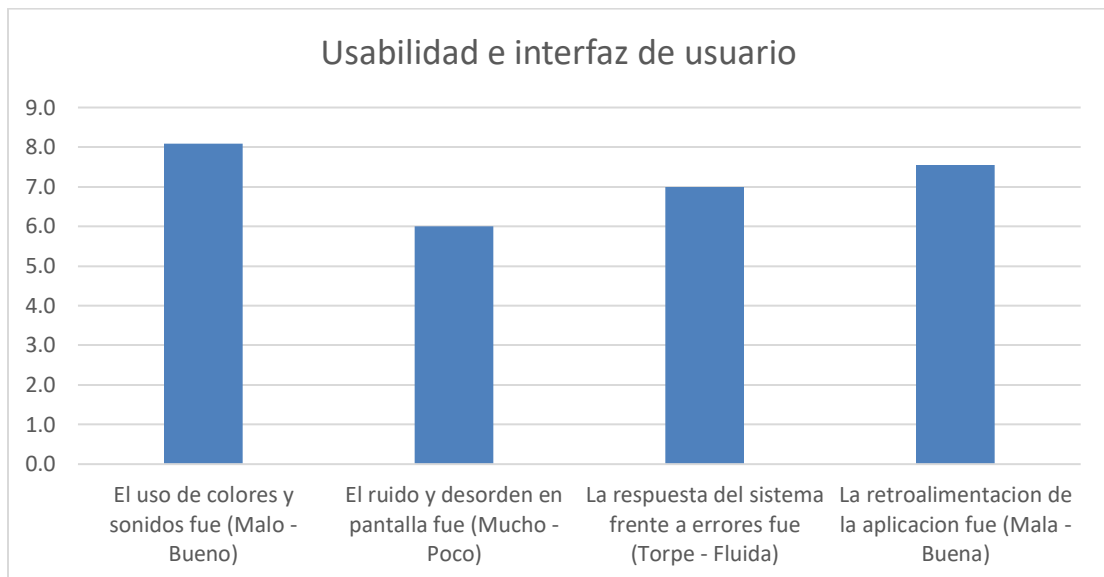


Ilustración 20 Grafico de valores promedio a las respuestas de la encuesta de usabilidad e interfaz de usuario.

A pesar de que los resultados fueron en general altos, se puede observar que existe una tendencia hacia puntajes más bajos en las preguntas relacionadas a la facilidad de uso de la aplicación, la organización de los elementos en pantalla y la fiabilidad de esta. La aplicación no cuenta con un tutorial como si lo hacen muchos juegos, y a pesar de que nosotros no consideramos la aplicación un juego como tal, la inclusión de uno podría ayudar bastante para aliviar la curva de aprendizaje inicial de esta. Respecto a la fiabilidad esto se puede deber a que hubo momentos durante las pruebas en donde se tuvo que realizar cambios al *backend* y otros en los que este no estuvo

activo, pero esto también pudo ser causado por errores en la aplicación que fueron relacionados a la descarga de información del *backend* que fueron corregidos después.

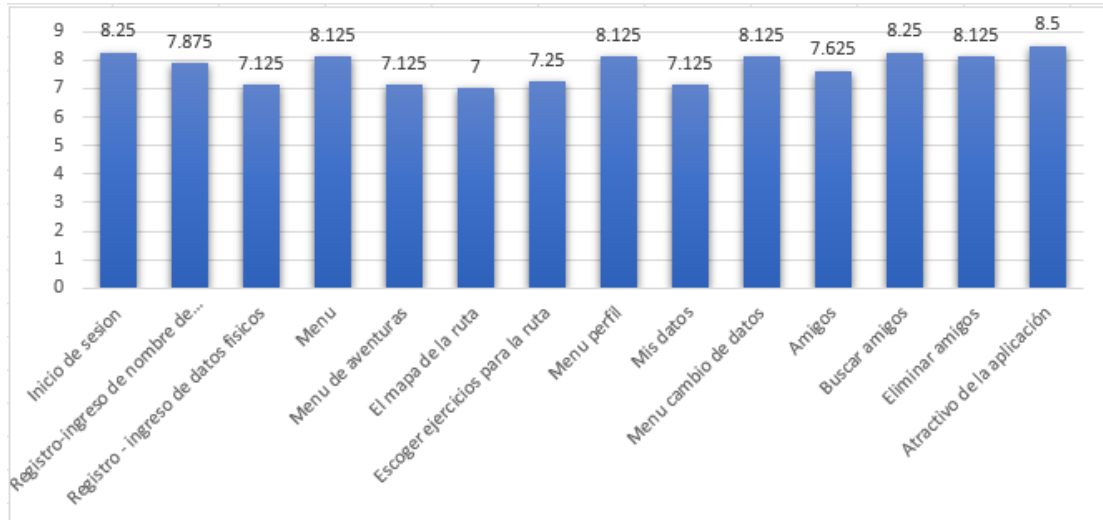


Ilustración 21 Grafico de valores promedio a las respuestas de la encuesta de valor estético

En general la aplicación tuvo un fuerte recibimiento en lo que corresponde a lo atractivo que fue con una puntuación de 8.5 siendo 9 lo más alto. En el caso de la pregunta de ¿qué tan intuitivo es la aplicación? (por página) los resultados totales estuvieron entre 7.0 y 8.25 lo que garantiza que cada pantalla es fácil de aprender y que con esto la aplicación cumple con su objetivo de ser atractivo y fácil de usar. Más a detalle en las pantallas la pantalla de Mapa de la ruta obtuvo la puntuación más baja con 7.0 hablando con algunos encuestados explicaron que la temática de fantasía que se trató y la del mapa (que es el mapa base de Android) genera discordia. La estrategia que se intentó en esta pantalla fue la de agregar elementos medievales que ayuden a esa inmersión como antorchas como marcadores y estandartes medievales como bordes. Sin embargo, más allá de lo que se propuso lo que se espera de esa pantalla para alcanzar el siguiente nivel satisfacción es la edición completa del mapa con la temática de la aplicación y por ende quedaría como sugerencia si se desea continuar el proyecto.

10 Compromiso de apoyo de la Institución

Durante el desarrollo de este proyecto se plantea hacer uso de los conocimientos de algunos departamentos de la universidad mediante asesorías en temas como la salud y el ejercicio, para ello haremos uso de las asesorías profesionales del centro deportivo javeriano. Con esto buscamos poder plantear ejercicios para nuestra aplicación que le permitan al usuario disfrutar de esta mientras se ejercita, pero, teniendo en cuenta que no se busca hacer que el usuario se sobre esfuerce en la realización de los ejercicios planteados. De igual manera haremos uso de las asesorías brindadas por nuestro director de tesis para poder desarrollar la mejor aplicación posible haciendo uso de las herramientas y técnicas más afines a la solución planteada y a la metodología SCRUM

11 Derechos patrimoniales

La autoría de este trabajo de grado pertenece al grupo de elaboración de este y el organizador, recaudador y plasmador. Suponiendo que los responsables de la elaboración de este proyecto autoricen a un tercero para que su distribución (aplicación). La autorización concedida cubre la modificación, distribución, comercialización y acceso completo al código fuente de la aplicación y de igual manera a los aspectos audiovisuales de la misma.

12 Conclusiones

Para esta tesis se planteó como objetivo incentivar a la gente a realizar ejercicio o como un mínimo salir de sus casas a caminar en orden de trabajar el sedentarismo en la población bogotana con ciudadanos con edad entre 18 y 25 años. Este objetivo fue cumplido con la aplicación en etapas tempranas del desarrollo, ya que se vio un amplio recibimiento de la solución implementada, como muestran las pruebas, lo que nos lleva a la conclusión que: ludificar el ejercicio mediante la adición de elementos multimedia y de videojuegos incentiva a la gente a practicarlo y es algo que consideramos se debe implementar con mayor frecuencia para atraer más personas a un estilo vida menos sentaría.

El reconocimiento de ejercicios fue un punto importante de nuestra aplicación ya que gracias a ellos se podría realizar los eventos propuestos sobre las rutas. Sin embargo, para cumplir nuestro objetivo no era necesario realizar una forma robusta de identificar los ejercicios, por ende, no se le dio mucho énfasis, de hecho, la forma de identificar los ejercicios fue mediante experiencia propia. No obstante, se sugiere mejorarlo mediante herramientas como la IA, que mejoraría este servicio al detectar más variedad de ejercicios y con mucha más precisión. Esto ampliaría la cantidad de ejercicios que se pueden trabajar en la aplicación siendo esto un agregado muy valioso para la solución.

La prueba de estética nos mostró que la solución creada es fácil de usar y atractiva a la vista, pero, aunque esto sea cierto el uso de la aplicación no puede depender solo de lo intuitivo ya que esta aplicación incluye muchos elementos extras que solo hacer un ejercicio o caminar en la calle. Estos elementos hacen que la aplicación presente los ejercicios como un videojuego y por ende necesitan un tutorial que muestre la dinámica de como propone la aplicación la realización de los ejercicios propuestos.

13 Referencias

- Alan, G. (2002). *Shared Fantasy: Role Playing Games as Social Worlds*. University of Chicago Press.
- Broom, D. R., Lee, K. Y., Lam, M. H., & Flint, S. W. (2019). Gotta catch 'em all or not enough time: Users motivations for playing Pokémon Go™ and non-users' reasons for not installing. *Health Psychology Research*, 7(1).
- Eoghan, E., Mackessy, T., Geary, C., Noonan, S., & Buckley, P. (marzo de 2019). An Empirical Study of Gamification Frameworks. *Journal of Organizational and End User Computing*, 31(1), 22-38.
- Google Developers. (08 de 03 de 2022). *Distribution dashboard*. (Google Developers) Recuperado el 23 de 05 de 2022, de <https://developer.android.com/about/dashboards>
- Google Developers Training team. (29 de 04 de 2022). *Download and install Android Studio*. (Google Developers) Recuperado el 23 de 05 de 2022, de Android Developers: <https://developer.android.com/codelabs/basic-android-kotlin-compose-install-android-studio#1>
- Hamari, j., & Koivisto, J. (agosto de 2015). Why do people use gamification services? *International Journal of Information Management*, 35(4), 419-431. Obtenido de <https://linkinghub.elsevier.com/retrieve/pii/S0268401215000420>
- Harper, P., & Norman, K. (1993). Improving user satisfaction: the questionnaire for user interaction satisfaction version 5. Obtenido de <https://www.semanticscholar.org/paper/Improving-user-satisfaction%3A-the-questionnaire-for-Harper-Norman/ab1430042d6b1a2f4b52087b74b0ddc0ea22cade>
- ISO 25000. (2022). *La familia de normas ISO/IEC 25000*. (ISO 25000) Recuperado el 22 de mayo de 2022, de <https://iso25000.com/index.php/normas-iso-25000>
- Kaczmarek, L. D., Misiak, M., Behnke, M., Dziekan, M., & Guzik, P. (2017). The Pikachu effect: Social and health gaming motivations lead to greater benefits of Pokémon GO use. *Computers in Human Behavior*, 75, 356-363. Obtenido de <https://www.sciencedirect.com/science/article/pii/S0747563217303527>
- Koivisto, J., & Hamari, J. (2014). Demographic differences in perceived benefits from gamification. *Computers in Human Behavior*, 35, 179-188. doi: 10.1016/j.chb.2014.03.007
- Marcus, A. (2014). *Design, user experience, and usability: third international conference, DUXU 2014, held as part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014 proceedings*. Cham, Suiza: Springer.

- Obesidad, un factor de riesgo en el covid-19.* (s.f.). Obtenido de <https://www.minsalud.gov.co/Paginas/Obesidad-un-factor-de-riesgo-en-el-covid-19.aspx>
- Obesity: preventing and managing the global epidemic. Report of a WHO consultation. (2000). *World Health Organization Technical Report Series*, 39.
- Saaty, M., Haqq, D., Toms, D. B., Eltahir, I., & McCrickard, D. S. (2021). A Study on Pokemon GO: Exploring the Potential of Location-based Mobile Exergames in Connecting Players with Nature. En *Extended Abstracts of the 2021 Annual Symposium on Computer-Human Interaction in Play* (págs. 128-132). New York, NY, USA: Association for Computing Machinery. Obtenido de <https://doi.org/10.1145/3450337.3483481>
- Shimizu, K., Shiko, Y., Kawasaki, Y., Orita, S., Inage, K., Shiga, Y., . . . Sato, M. (2021). Effects of Nintendo Ring Fit Adventure Exergame on Pain and Psychological Factors in Patients with Chronic Low Back Pain. *Games for Health Journal*, 10(3).
- Wang, A. I. (2021). Systematic literature review on health effects of playing Pokémon Go. *Entertainment Computing*, 38, 100-411. doi: 10.1016/j.entcom.2021.100411
- Yim, J., & Graham, T. (2007). Using games to increase exercise motivation. *the 2007 conference* (pág. 166). Toronto, Canada: ACM Press.

14 Anexos

Anexo 1 Plan de proyecto SPMP

Anexo 2 Documento de especificación de requisitos SRS

Anexo 3 Documento de especificación de diseño SDD

Anexo 4 Propuesta de trabajo de grado

Anexo 5 Documento de control de calidad