
La gamificación como herramienta para
fomentar y motivar la actividad física.
Gamification as a tool to promote and motivate
physical activity



Trabajo de Fin de Grado
Curso 2024–2025

Autores

Antonio Povedano Ortiz, Raúl Saavedra de la Riera

Directores

Antonio Calvo Morata
Cristina Alonso Fernández

Grado en Desarrollo de Videojuegos
Facultad de Informática
Universidad Complutense de Madrid

La gamificación como herramienta para
fomentar y motivar la actividad física.
Gamification as a tool to promote and
motivate physical activity

Trabajo de Fin de Grado en Desarrollo de Videojuegos

Autores

Antonio Povedano Ortiz, Raúl Saavedra de la Riera

Directores

Antonio Calvo Morata
Cristina Alonso Fernández

Convocatoria: *Junio 2025*

Calificación Antonio Povedano Ortiz: *10*

Calificación Raúl Saavedra de la Riera: *10*

Grado en Desarrollo de Videojuegos

Facultad de Informática

Universidad Complutense de Madrid

16 de junio de 2025

Dedicatoria

*A mis padres, por siempre confiar en mí y estar ahí para todo lo que me proponga. A mis abuelos, Paco e Isabel: aunque no hayan podido verme, sé que estarían orgullosos de mí. Siempre los llevaré conmigo.- **Antonio***

*A mi perra por molestarme. A mi padre, por cada consejo. A Chanita y Nuria por no dejarme ir. A Pepín y Toni, por aguantarme tantos años. A Yaiza y Manu, por siempre decirme que soy increíble aunque sea mentira. A mi abuelo Manolo, que aunque no pueda verlo, me daría el "positivo". Y, aunque no quiera, a mi madre, que me ha apoyado en cada tontería. - **Raúl***

Agradecimientos

Queremos dar las gracias a nuestros tutores, Toni y Cristina, por hacer que este trabajo haya sido mucho más llevadero. Gracias por estar siempre ahí para echarnos una mano y resolver todas nuestras dudas.

También gracias a todas las personas que participaron en nuestras pruebas, que colaboraron con sus ideas, sugerencias y tiempo. Nos habéis ayudado a que este proyecto saliera adelante.

Resumen

La gamificación como herramienta para fomentar y motivar la actividad física.

Durante mucho tiempo, los videojuegos se han visto como una actividad sedentaria e incluso dañina para la salud física y psicológica de los jugadores. Se ha argumentado que fomentan el sedentarismo, que generan adicción y afectan directamente al bienestar emocional. Sin embargo, los videojuegos han demostrado en numerosas ocasiones sus efectos positivos en diversos ámbitos, como la educación, el tratamiento terapéutico o en promocionar los hábitos saludables. Entre estos, destacan los exergames, videojuegos en los que el movimiento físico es esencial para jugar, con títulos como *Pokemon Go* o *Zombies, Run!*. Al requerir de actividad física, han convertido el ejercicio en una experiencia más atractiva y accesible para todo tipo de personas, aumentando la motivación para realizar ejercicio.

Este trabajo ha consistido en desarrollar un videojuego en línea que incentive la actividad física en los jugadores y estudiar los efectos que tiene en los hábitos de los jugadores a través de pruebas con usuarios.

Los análisis de los datos recogidos durante las pruebas muestran que existe una mejora de los hábitos de ejercicio de los usuarios a corto plazo, además de la buena aceptación al uso de este durante sus rutinas de ejercicio. Para permitir realizar nuevos estudios a largo plazo, el juego se ha desarrollado con una arquitectura que permite introducir nuevos elementos dentro del juego de una manera sencilla, para posteriormente validar sus efectos.

Palabras clave

Gamificación, Hábitos saludables, Actividad física, Videojuegos

Abstract

Gamification as a tool to promote and motivate physical activity

For a long time, video games have been seen as a sedentary activity and even harmful to the physical and mental health of players. It has been said that they promote inactivity, create addiction, and have a negative effect on emotional well-being. However, video games have also shown many positive effects in different areas, such as education, therapy, or promoting healthy habits. One example is exergames, which are video games where physical movement is necessary to play. Games like Pokémon Go or Zombies, Run! make exercise more fun and accessible for many people, increasing their motivation to be active.

This project focused on developing an online video game that encourages physical activity in players and studying the effects it has on their exercise habits through user testing.

The data collected during the tests shows an improvement in the users' exercise habits in the short term, as well as a good acceptance of the game as part of their workout routines. To allow future long-term studies, the game was built with an architecture that makes it easy to add new features later and evaluate their effects.

Keywords

Gamification, Healthy habits, Physical activity, Video games

Índice

1. Introducción	1
1.1. Motivación	1
1.2. Objetivos	2
1.3. Plan de trabajo	2
2. Estado de la Cuestión	5
2.1. Contextualización del problema	5
2.2. Los exergames	6
2.2.1. Juegos comerciales para dispositivos móviles	7
2.2.2. Juegos comerciales para otros dispositivos	10
2.2.3. Juegos orientados a la investigación	12
2.3. Nuestra propuesta	14
3. Descripción del Trabajo	17
3.1. Diseño general de la propuesta	17
3.1.1. Premisa inicial: los critterons	17
3.1.2. Enfoque de la aplicación	17
3.1.3. Experiencia de juego	18
3.1.4. Ciclo de juego	19

3.1.5.	Progreso y retorno de jugador	21
3.1.6.	Factor social	22
3.2.	Entorno de desarrollo para el cliente	23
3.2.1.	Candidatos potenciales	24
3.2.2.	Herramienta de desarrollo general o especializada en videojuegos	26
3.2.3.	Selección del entorno final donde desarrollaremos la aplicación	28
3.3.	Desarrollo del cliente en Unity	31
3.3.1.	Hitos durante el desarrollo	31
3.3.2.	Estructura del proyecto	33
3.3.3.	Principales clases, modularidad y escenas	33
3.3.4.	Assets y addons usados para el proyecto	38
3.3.5.	Geoposicionamiento y su integración	39
3.3.6.	Integración del podómetro	40
3.3.7.	Integración de la API en Unity y optimización de recursos . .	41
3.3.8.	Integración de Xasu	42
3.3.9.	Pulsera EmotiBit	43
3.4.	Desarrollo del servidor	45
3.4.1.	Estructura de los datos	45
3.4.2.	Almacenamiento de datos: MongoDB	47
3.4.3.	API y backend: Spring Boot	47
3.4.4.	Seguridad	49
3.4.5.	Automatización y despliegue	49
3.5.	Diseño y estudio del plan de pruebas	50
3.5.1.	Diseño del estudio y selección de participantes	50
3.5.2.	Objetivos del plan de pruebas	50
3.5.3.	Metodología de la prueba	51

3.5.4.	Métricas e interpretación de resultados	52
3.5.5.	Implementación de los cuestionarios y herramientas de seguimiento	52
3.6.	Desarrollo y resultados de las pruebas	53
3.6.1.	Grupo de participantes	53
3.6.2.	Desarrollo de la prueba semanal	60
3.6.3.	Resultados de las métricas del juego	60
3.6.4.	Resultados de la entrevista y el cuestionario final	70
3.6.5.	Resultados de los cuestionarios diarios	75
4.	Conclusiones y Trabajo Futuro	79
4.1.	Conclusiones	79
4.2.	Trabajo futuro	80
4.3.	Motivation	83
4.4.	Objectives	83
4.5.	Work Plan	84
4.6.	Conclusions	87
4.7.	Future Work	88
5.	Contribuciones Personales	91
	Bibliografía	95

Índice de figuras

2.1. Pokémon Go	8
2.2. Walkr: Fitness Space Adventure	9
2.3. Pocket Plants	9
2.4. Zombies, Run!	10
2.5. The Biggest Loser	11
2.6. Yourself! Fitness	12
3.1. Navegación entre pantallas	21
3.2. Escena del hotel. Izquierda vista general, derecha seleccionado un critteron	34
3.3. Escena del Combate	35
3.4. Pantallas en social. Izquierda lista de amigos, derecha ranking de amigos	35
3.5. Escena del estadísticas de usuario	36
3.6. Escena principal. Izquierda vista general, derecha seleccionada una parada	37
3.7. Escena de login. Izquierda inicio de sesión, derecha creación de una cuenta.	37
3.8. Ejemplo mapa	39
3.9. Estructura de la base de datos	46
3.10. Sexo de los usuarios	53

3.11. Edad de los usuarios	54
3.12. Frecuencia de ejercicio de los usuarios	54
3.13. Tiempo de ejercicio de los usuarios	55
3.14. Tipo de ejercicio realizado por los usuarios	55
3.15. Ejercicios de cardio realizado por los usuarios	56
3.16. Cómo realizan el ejercicio los usuarios	56
3.17. Dónde realizan el ejercicio los usuarios	57
3.18. Motivación por hacer ejercicio de usuarios	58
3.19. Motivo para realizar ejercicio de los usuarios	58
3.20. Usuarios que usan tecnologías durante los ejercicios	59
3.21. Usuarios que han jugado a exergames	59
3.22. Pasos globales de los usuarios	62
3.23. Incremento de los pasos por día de los usuarios	62
3.24. Niveles de los usuarios	63
3.25. Incremento de los niveles por día de los usuarios	64
3.26. Número de critterons de los usuarios	64
3.27. Números de habitaciones de los usuarios	65
3.28. Tipos de combates realizados por los usuarios	66
3.29. Victorias y derrotas de los usuarios	66
3.30. Uniones de amistad entre los usuarios	67
3.31. Número de veces que se ha obtenido cada critteron	68
3.32. Ejemplo de rutas y recorridos de forma lúdica	69
3.33. Ejemplo de desplazamientos en un entorno profesional o doméstico	69
3.34. Frecuencia de ejercicio de los usuarios tras la semana de pruebas	71
3.35. Tipo de ejercicio realizado por los usuarios tras la semana de pruebas	71
3.36. Dónde realizan el ejercicio los usuarios tras la semana de pruebas	72

3.37. Cómo realizan el ejercicio los usuarios tras la semana de pruebas . . .	72
3.38. Cuánto tiempo le dedican al ejercicio los usuarios tras la semana de pruebas	73
3.39. Usuarios que usan tecnologías durante los ejercicios tras la semana de pruebas	74
3.40. Tipos de tecnologías usadas tras la semana de pruebas	74
3.41. Usuarios que juegan exergames tras la semana de pruebas	75
3.42. Número de usuarios que realizaron ejercicio por día	76
3.43. Qué tipo de ejercicio realizan los usuarios	76
3.44. Cuánto ejercicio hacen los usuarios	77
3.45. Dónde realizan ejercicio los usuarios	77
3.46. Uso de dispositivos/aplicaciones por día por los usuarios	78

Índice de tablas

3.1. Comparativa de motores y frameworks según nuestras necesidades . .	26
3.2. Comparación entre entornos generalistas y motores de videojuegos . .	27
3.3. Resumen de ventajas y desventajas de los diferentes motores	30
3.4. Características de los usuarios	61

Introducción

“El paso más importante que puede dar un hombre. No es el primero, ¿verdad? Es el siguiente. Siempre el siguiente paso, Dalinar.”

— Brandon Sanderson

1.1. Motivación

Una de las principales motivaciones para realizar este proyecto es demostrar que los videojuegos no tienen por qué fomentar el sedentarismo, sino que también pueden ser una herramienta eficaz para promover la actividad física y hábitos saludables. Esta visión le da al proyecto un valor añadido, ya que combina el entretenimiento con un propósito positivo. Además, nos brinda la oportunidad de explorar un campo que nos interesa desde un inicio: el uso de la tecnología para generar un impacto real en el bienestar de las personas.

Por otro lado, nos presenta una oportunidad para poder aplicar todos los conocimientos que adquirimos durante la carrera. De esta manera, la posibilidad de crear un videojuego en línea, así como el desarrollo de las estructuras de datos y los proyectos necesarios para poder desplegarlo, presenta un reto muy atractivo. Además, este proyecto no solo nos permite usar lo aprendido, sino que también nos permite explorar herramientas y disciplinas con las que no trabajamos previamente, haciendo que el proyecto sea aún más interesante. Entre estas se incluyen la creación de nuestra propia API, siendo esta el conjunto de llamadas y protocolos que nos permiten comunicarnos con los datos, así como el uso de bases de datos, entre otras.

Por último, y con relación a lo anterior dicho, realizar este proyecto nos ayuda personalmente a aprender y demostrar nuestros conocimientos de una manera más clara, logrando al finalizar un producto tangible, lo cual es fundamental para nuestro

futuro profesional.

1.2. Objetivos

El objetivo principal de este proyecto es la creación de un videojuego para promover la actividad física y estudiar sus efectos en los hábitos de los jugadores. Para poder cumplir este objetivo, se proponen una serie de objetivos parciales:

1. Realizar un estudio del estado actual del campo de los juegos que fomentan la actividad física. De esta manera se tendrá una idea más clara de lo que ya se está usando y qué técnicas se están empleando para motivar a los jugadores a realizar más ejercicio.
2. Crear un diseño inicial del videojuego. En este punto también está la realización de un estudio para determinar qué entorno de desarrollo se adapta mejor a las necesidades del proyecto.
3. Desarrollar el videojuego una vez comprendida la visión y alcance general del proyecto. Se incluye tanto la parte visual del juego como el sistema que gestiona los datos.
4. Diseñar un plan de pruebas para recoger datos de los jugadores de manera controlada y llevarlo a cabo.
5. Analizar los datos recogidos durante las pruebas e identificar mejoras a realizar.

1.3. Plan de trabajo

Como se ha descrito anteriormente, al haber una gran cantidad de objetivos que se han tenido que ir completando, la organización del proyecto ha sido una parte fundamental para que este se haya completado en el tiempo esperado y haya tenido la mayor calidad posible.

Para ello, se han organizado reuniones cada dos semanas con los tutores del proyecto, con el objetivo de planificar el trabajo, hacer un seguimiento de las tareas y resolver dudas. Además, el trabajo se ha dividido en hitos que han abordado cada uno de los objetivos parciales mencionados. Una vez que se han completado todos estos hitos, se ha dado por concluido el proyecto. Para la gestión y el control del desarrollo, se ha utilizado GitHub como sistema de control de versiones. El repositorio del proyecto ha estado disponible en este enlace: <https://github.com/RaulSaavedraRiera/TFG2024-AntonioPRaulS>

Estudio y diseño inicial

El principal objetivo ha sido asentar las bases del proyecto. Por ello, primero se ha realizado una búsqueda de los juegos que fomentan la actividad física. Además, se ha llevado a cabo una búsqueda de artículos científicos para ver cómo se han empleado en el mundo de la investigación. Se ha tenido que hacer una documentación donde se ha guardado la información más relevante de estos. Tras esto, se ha realizado la creación de un Documento de Diseño de Juego con los puntos claves del videojuego a desarrollar. Ligado a esto, se ha hecho un estudio sobre qué entornos de desarrollo se han ajustado más a las necesidades del videojuego. Además, al ser un videojuego en línea, uno de los principales retos ha sido cómo almacenar y acceder de una manera óptima y estable a los recursos. Para ello, se ha tenido que realizar otro estudio enfocado en definir qué tipo de datos han sido necesarios y cómo han sido representados. De esta manera se ha realizado el diseño del sistema servidor para poder acceder a estos recursos de una manera óptima y ver cuál ha sido el entorno más eficiente para desarrollarlo. Debido a la importancia de entender el proyecto desde un inicio, para la realización de este bloque se ha establecido la duración de un mes para completarlo.

Desarrollo del videojuego

El objetivo de este bloque ha sido la implementación del propio videojuego, así como el desarrollo de la infraestructura necesaria para gestionar la comunicación entre el videojuego y el servidor. Para ello, se han realizado pruebas para comprobar la viabilidad del sistema de almacenamiento de datos y su integración. Una vez completadas con éxito estas pruebas, se ha procedido al desarrollo de la lógica del servidor, incluyendo la conexión con la base de datos y la implementación de las funcionalidades necesarias para enviar y recibir información desde el videojuego. A medida que se han ido incorporando nuevas funcionalidades al videojuego, estas se han conectado con el servidor para validar su correcto funcionamiento. Finalmente, se ha trabajado en la automatización del proceso de despliegue, permitiendo ejecutar el servidor en distintos entornos de forma sencilla y guiada. Para ello, se han desarrollado dos proyectos: uno destinado a automatizar la subida de datos y otro enfocado a facilitar el despliegue multiplataforma. El tiempo estimado para completar esta fase ha sido de seis meses.

Pruebas con usuarios

En este bloque se ha desarrollado el diseño experimental, que ha incluido el guión de las pruebas con usuarios y los diferentes cuestionarios que han tenido que responder. A su vez, ha incluido todo el seguimiento y ayuda necesarios durante la realización de las pruebas y la recolección de datos durante estas, para posteriormente haber sido procesados. El tiempo para completar esta fase ha sido de un

mes.

Análisis de datos

Este ha sido el último bloque de trabajo. Una vez concluidas las pruebas y obtenida toda la información posible, se han procesado dichos datos para poder extraer información más precisa que ha permitido sacar diferentes conclusiones con las que se ha establecido una postura final sobre el éxito del proyecto. El tiempo para completar esta fase ha sido de un mes.

Capítulo 2

Estado de la Cuestión

La relación entre los videojuegos y la actividad física no es evidente. Son conceptos tradicionalmente separados e incluso percibidos como opuestos. Sin embargo, el número de videojuegos que contradicen esto ha aumentado en gran medida e incluso consolas tan famosas como la *Wii* basan su propuesta en la actividad física.

Por ello en este apartado recogeremos algunas de esas propuestas que han tratado de unir ambos mundos: explorando su alcance, sus limitaciones y el contexto en el que fueron creadas. Este análisis permite identificar qué aportan cada una de ellas, qué espacios quedan por cubrir y, por tanto, qué podemos aportar con nuestra propuesta.

2.1. Contextualización del problema

En la sociedad actual, la gran variedad de actividades de ocio, la gran mayoría de ellas digitales y sedentarias, ha hecho que, para muchas personas, la actividad física quede en segundo plano. Aunque hay quienes practican ejercicio regularmente, otras opciones como ver series o consumir contenido en línea cada vez más ocupan más parte de nuestro tiempo libre. Además, gran parte de los trabajos actuales requieren poca o nula actividad física. Precisamente, es en el ámbito del ocio, los videojuegos han experimentado un gran crecimiento, pasando de ser una afición de nicho se ha convertido en una forma de ocio masiva. Solo en España durante 2024 se alcanzaron los 22,1 millones de jugadores, lo que representa aproximadamente el 45 % de la población (Asociación Española de Videojuegos (AEVI), 2024). Además ha conseguido que esta industria facture más que la del cine y la música juntas (Harris, 2024).

Muchas personas ven a los videojuegos como un enemigo de las actividades más saludables y principal causante del sedentarismo. Sin embargo, en los últimos años,

junto a la normalización de este medio de entretenimiento, ha surgido una nueva corriente llena de propuestas innovadoras que muestra cómo los videojuegos no solo no son incompatibles con la actividad física, sino, una actividad que puede dar un mayor atractivo al ejercicio y atraer a personas que, de otra manera, nunca se lo habrían propuesto (Staiano y Calvert, 2011; Gao et al., 2015).

2.2. Los exergames

Para entender de una mejor forma el objetivo de este proyecto tenemos que hablar primero de los juegos serios. Estos están diseñados con un propósito que va más allá del ocio, enfocados en ámbitos como la educación, la capacitación, la investigación o el tratamiento de enfermedades, entre muchos otros. Una de sus principales ventajas es que colocan al usuario en un rol activo, provocando que el jugador se involucre, tome decisiones y realice acciones para avanzar. Además, rompen barreras de atención y favorecen una mayor implicación de los jugadores en la tarea que fomentan.

Para lograr esto, los juegos serios aplican mecánicas y dinámicas propias de los juegos. Es decir, utilizan estrategias propiamente lúdicas con el objetivo de hacer actividades complejas o monótonas mucho más motivadoras. Esta estrategia es conocida como gamificación. La gamificación consiste en trasladar los elementos propios de los juegos (como los retos o las recompensas) a otros contextos como la educación, la productividad o la salud, consiguiendo con ello un mayor compromiso por parte de los usuarios (Spinify, n.d.). Por ejemplo, una tarea aburrida a priori para los niños, como puede ser recoger los juguetes, se puede volver más llevadera y entretenida si se esconde detrás de un juego. Encontramos múltiples ejemplos de esto en la educación y el ámbito empresarial, donde se han obtenido resultados muy positivos (Werbach y Hunter, 2012).

En muchos casos, la frontera entre una aplicación gamificada y un videojuego como tal puede llegar a diluirse. Un ejemplo concreto de esto es *Duolingo*. *Duolingo* es una aplicación para aprender idiomas que ha logrado captar el interés de millones de personas para realizar una tarea que, sin la capa de gamificación que usa, podría resultar pesada, como es la de estudiar idiomas. Gracias a su planteamiento, donde se combinan elementos como rankings, juegos integrados en las lecciones y rachas que puedes perder si no practicas día a día, han logrado construir una comunidad sólida que no solo aprende, sino que se entretiene usando la aplicación (Deci et al., 1991). Según a quién se le pregunte, *Duolingo* puede considerarse un juego en sí mismo o una aplicación gamificada. Otro aspecto importante es que la gamificación puede funcionar aún mejor si se le incluye un enfoque competitivo entre los participantes, logrando que no se centren tanto en la tarea que realizan, sino en obtener mejores resultados que sus compañeros y destacar respecto a los demás (Herrmann et al., 2019).

Al entender el funcionamiento y las ventajas de los juegos serios y la gamificación,

resulta natural aplicar este enfoque al ámbito de la actividad física. De ahí surgen los exergames. Los exergames son videojuegos diseñados para promover la actividad física. A diferencia de los videojuegos tradicionales, en los que la interacción suele estar limitada a mandos o teclados, los exergames requieren que el jugador utilice su cuerpo para ejecutar acciones dentro de éste.

Tras entender qué son los exergames y la principal estrategia que utilizan, se analizarán los principales juegos más conocidos para intentar entender qué ha causado su éxito, además de identificar qué herramientas utilizan para motivar el ejercicio en sus jugadores. Muchos de los juegos de los que se hablará más adelante han sido extraídos de Wattanasoontorn et al. (2013).

2.2.1. Juegos comerciales para dispositivos móviles

En este apartado se analizarán una serie de juegos de móvil de carácter comercial. Aunque incorporan elementos que promueven la actividad física, su diseño y desarrollo están principalmente orientados a su rentabilidad económica. Por este motivo, muchos de estos títulos incluyen funciones exclusivas para usuarios suscritos o la posibilidad de realizar microtransacciones, que permiten obtener mejoras y recompensas de forma más rápida.

2.2.1.1. Pokémon Go

Uno de los títulos más conocidos y que mayor impacto ha generado en el sector es *Pokémon Go* (Figura 2.1). Fue uno de los primeros exergames que tuvo repercusión a nivel mundial, llegando a tener más de 80 millones de jugadores al mes. Es un juego de realidad aumentada que, usando el GPS, nos permite capturar, luchar y participar en eventos con sus famosas criaturas. Protagonistas de una de las mayores franquicias de la compañía *Nintendo* y desarrollado por *Niantic*.

Entre las distintas estrategias que emplea este juego para fomentar el ejercicio físico en los jugadores, se pueden destacar la aparición de distintos tipos de Pokémon dependiendo de la ubicación, lo cual motiva a los jugadores a desplazarse físicamente para capturarlos; la mecánica de incubación de huevos, los cuales abren tras recorrer una determinada cantidad de kilómetros; la defensa y conquista de gimnasios, que requiere acudir a ubicaciones concretas para participar en combates; la recogida de objetos en *Poké Paradas*, que son puntos de interés reales a los que se debe acudir para obtener recursos dentro del juego; y el intercambio de *Pokémon*s entre jugadores, fomentando así la cooperación y socialización, ya que dicha interacción solo es posible cuando los jugadores se encuentran próximos físicamente. Sin embargo, no se puede afirmar con certeza cuál es el verdadero motivo de su éxito ni la eficacia real de las mecánicas utilizadas para incentivar el ejercicio, ya que el fuerte peso mediático de la franquicia y el fanatismo que la rodea pueden influir de manera determinante.

Al igual que este juego, existen otros títulos cuya forma de fomentar la actividad física es similar y cuyo principal atractivo radica también en la franquicia a la que pertenecen. Entre estos podemos encontrar títulos como *Jurassic World: Alive*. También en el juego *Harry Potter: Wizards Unite*, el cual cerró en enero de 2022. También en el juego podemos observar que las principales diferencias que se encuentran tienen que ver más con la tematización y cambios en el diseño, para poder adaptarse a la franquicia que usa (Laato, 2020).



Figura 2.1: Pokémon Go

2.2.1.2. Walkr: Fitness Space Adventure

Walkr: Fitness Space Adventure (Figura 2.2) es un juego que promueve la actividad física a través del descubrimiento y exploración espacial de un universo ficticio. Para ello, el juego muestra el tiempo restante para descubrir nuevos planetas o nuevas galaxias. Sin embargo, este tiempo se reduce cada vez que el jugador recorre una cierta distancia. Además, ofrece una visualización de las calorías quemadas a través del uso del juego, lo que contribuye a que el jugador perciba beneficios más allá del aspecto lúdico, reforzando la utilidad del juego en su salud.

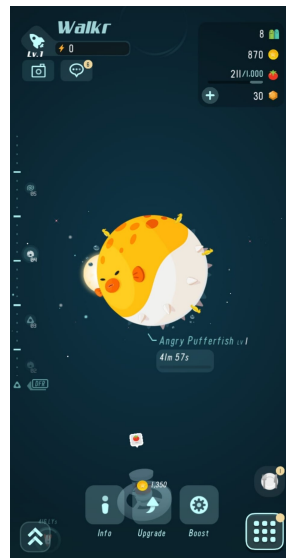


Figura 2.2: Walkr: Fitness Space Adventure

2.2.1.3. Pocket Plants

Pocket Plants (Figura 2.3) se basa en la gestión de un pequeño jardín, en el que el jugador debe ir desbloqueando y combinando diferentes tipos de plantas para obtener nuevas. La motivación principal para hacer ejercicio es que cuando el jugador camina obtiene dentro del juego monedas, que le permite reducir los tiempos de espera necesarios para realizar las combinaciones entre plantas.

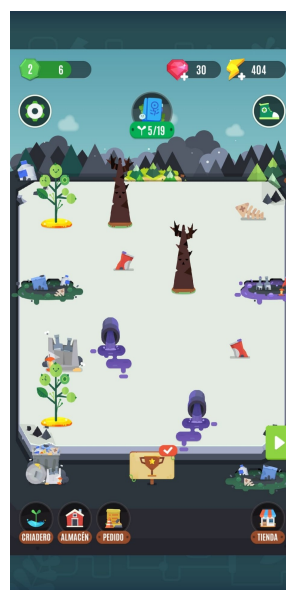


Figura 2.3: Pocket Plants

2.2.1.4. Zombies, Run!

Zombies, Run! (Figura 2.4) es un juego ambientado en un apocalipsis zombi. Para motivar la actividad física, el juego presenta una serie de misiones en las cuales serán necesarios ciertos objetos. Para obtenerlos se tendrá que caminar una cierta distancia. Además, se plantean los ataques de zombis, que son eventos inesperados que obligan al jugador a correr durante un breve periodo de tiempo para evitar ser atrapado por estos. Introduce también distintos modos de juego como el modo historia, el modo entrenamiento o el modo carrera, promoviendo un mayor uso de la aplicación y con esto establecer un hábito de ejercicio en el jugador.

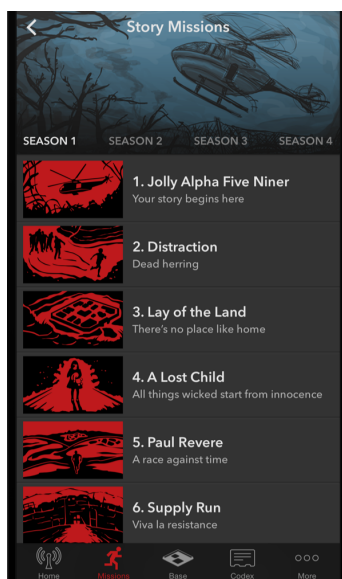


Figura 2.4: *Zombies, Run!*

Cabe destacar que aunque muchos de estos títulos han logrado amasar una gran cantidad de jugadores y hacer que se muevan para completar los diferentes objetivos que plantea el juego, su enfoque comercial quita cierta importancia al ejercicio, ya que priorizan otros métodos para lograr cumplir las tareas que propone, habitualmente relacionados con la monetización del juego.

2.2.2. Juegos comerciales para otros dispositivos

Además de en dispositivos móviles podemos encontrar exergames en diferentes dispositivos como las consolas. En estos el movimiento suele ser algo más estático y suelen centrarse en ejercicios de cardio o estiramientos. Aunque también incluyen elementos que promueven la actividad física, su propósito principal continúa siendo comercial.

2.2.2.1. The Biggest Loser

The Biggest Loser (Figura 2.5) es un juego lanzado para la consola *Wii*, basado en el programa de televisión del mismo nombre, en el que los concursantes competían para perder peso. El videojuego usa esa dinámica utilizando los mandos de la consola, que permiten una gran movilidad, para realizar ejercicios interactivos.

Para incentivar el ejercicio entre los usuarios, el juego emplea varias estrategias, entre ellas: aprovechar la popularidad previa del programa, incluyendo personajes y entrenadores reconocidos; permitir la creación de entrenamientos personalizados adaptados a las necesidades de cada jugador; integración de mini-juegos; y mostrar el progreso del jugador a lo largo del tiempo.



Figura 2.5: The Biggest Loser

2.2.2.2. Yourself! Fitness

Yourself!Fitness (Figura 2.6) es un juego lanzado en ordenador y en la consola *Xbox 360*. Durante el juego nos guiará Maya, una entrenadora personal, a través de rutinas de entrenamiento que se ajustan en función del rendimiento y estado físico del jugador.

Aunque inicialmente no hacía uso del *Kinect*, el cual es un dispositivo que permite interactuar con una consola o ordenador sin necesidad de usar un mando, se acabó integrando en este. De esta manera se mejoró la interacción y aumentó el nivel de realismo y precisión del juego.



Figura 2.6: Yourself! Fitness

Como estos títulos, existen otros videojuegos con objetivos y mecánicas similares que también han tenido impacto en el sector del exergaming, como *Wii Fit*, *Ring Fit Adventure* o *Just Dance*, entre muchos otros. Todos ellos han contribuido a consolidar una categoría de juegos que han fomentado la incorporación de la actividad física en los videojuegos.

2.2.3. Juegos orientados a la investigación

En este apartado se analizarán una serie de juegos desarrollados con fines de investigación, cuyo objetivo principal es estudiar el impacto de los exergames en diversos aspectos.

2.2.3.1. Tito Bico

Tito Bico es un juego que utiliza el *Wii Balance Board*, un periférico con forma de báscula que contiene sensores de presión que permiten medir el peso del jugador y su distribución corporal, como forma de control para los personajes del juego. El objetivo de este es entregar paquetes de comida a través de diversas islas. Además fomenta otros hábitos saludables como el llevar una dieta equilibrada. Durante el juego se pueden ver mensajes informativos con curiosidades relacionadas con la nutrición (Scarle et al., 2011).

Este juego demuestra cómo los exergames pueden convertirse en herramientas dentro de otros objetivos, como en este caso, la importancia de tener una dieta equilibrada. Sin embargo, como se dice en el estudio, no se puede saber su eficacia ya que se requiere una implementación prolongada con usuarios reales. Aun así, las primeras pruebas muestran que estos juegos resultan atractivos para los niños, lo cual refuerza su potencial como recurso educativo. Estas pruebas, al estar orientado el juego a un público infantil, se realizaron en escuelas con 17 niños entre 7 y 11

años. La metodología que usaron incluyó cuestionarios antes de jugar para evaluar conocimientos previos, sesiones de juego supervisadas y entrevistas para recoger feedback sobre la experiencia y la dificultad del juego. El objetivo era entender cómo percibían el juego los niños y recoger propuestas de mejora directamente del público objetivo.

2.2.3.2. Fitness Adventure

Fitness Adventure es un juego para dispositivos móviles. El juego narra una historia ficticia mientras el jugador camina, que presenta información sobre lugares reales que se va encontrando durante su recorrido.

Además, para potenciar la inmersión del juego, se incorporaron etiquetas visuales, imágenes que pueden ser escaneadas para mostrar más información, y RFID, dispositivos que se pueden leer por radiofrecuencia para detectar ubicación o interacción con estos. Sin embargo, el uso de tecnología RFID resultaba poco práctico debido a la necesidad de que el dispositivo móvil contara con un lector compatible, algo poco común en ese momento (T. Fujiki, 2009).

Este proyecto se desarrolló en un momento en el que el concepto de exergame aún no era muy conocido por el público general. Sin embargo, el estudio afirma que los exergames ofrecen una alternativa para motivar la actividad física a través del juego.

2.2.3.3. Heart Angel

Heart Angel es un juego para dispositivos móviles. Su principal mecánica combina el ejercicio físico con una dinámica de combate inspirada en el juego *Space Invaders*. Los jugadores deben eliminar virus invasores mientras realizan actividad física y alcanzan ciertos niveles de frecuencia cardíaca. Para ello, utilizan un monitor de ritmo cardíaco conectado mediante Bluetooth.

A medida que los jugadores alcanzan ciertos niveles de frecuencia, el juego ofrece bonificaciones como escudos o bombas que les ayudan a avanzar (Yim y Moffat, 2009).

El estudio mostró que es importante equilibrar el esfuerzo físico del jugador con la forma en que interactúa con el juego, porque así el ejercicio se siente más divertido y menos cansado. Para ello, las pruebas que realizaron se centraron en la relación entre el esfuerzo físico y la dificultad del juego, utilizando dispositivos de medición como monitores de frecuencia cardíaca para adaptar el nivel del juego en función del rendimiento físico del usuario. También se realizaron entrevistas posteriores para valorar la experiencia global.

2.2.3.4. Pinizoro

Pinizoro es un juego dirigido especialmente a niños de entre 4 y 12 años, donde los jugadores asumen el rol de una fuerza secreta encargada de combatir piratas, zombis, ninjas y robots en su vecindario. Mediante el uso de mapas creados por otros jugadores, deben recorrer rutas caminando y enfrentarse a enemigos mediante rompecabezas.

Una de sus principales características es el editor de mapas que permite diseñar rutas personalizadas para que otros jugadores las sigan. Esta mecánica no solo promueve el ejercicio físico, sino también la creatividad y el aprendizaje (Marcelo Coelho, 2010).

En las pruebas realizadas con familias, se observó una alta aceptación por parte de los padres, quienes valoraron positivamente el uso de la tecnología digital para fomentar caminatas. Por ello, se afirma que los exergames se pueden usar como una herramienta para motivar a familias que no suelen realizar actividad física juntas. Para las pruebas, como hemos podido ver, se contó con la opinión de padres y profesores para valorar la comprensión de los menús, la accesibilidad y la posibilidad de implementar el juego en casa o en aulas. Se hicieron entrevistas grupales que permitieron identificar problemas como el elevado coste del hardware o la claridad de los juegos. Se realizaron, además, sesiones individuales con cuatro participantes adultos que también crearon sus propios mapas. El objetivo era observar la usabilidad, la integración del ejercicio físico dentro del juego y el interés a medio-largo plazo.

2.3. Nuestra propuesta

Como hemos visto, la combinación de ejercicio y videojuego no es algo nuevo. Desde la generalización de los teléfonos inteligentes, gran cantidad de juegos y aplicaciones, con menor o mayor éxito, han surgido en torno a esta idea. Sin embargo, consideramos que aún se tiene que investigar y entender mejor los métodos que hacen que los jugadores realicen una mayor actividad física y el impacto real en ellos.

Al igual que se ha visto en los estudios, creemos que tenemos la posibilidad de generar una propuesta de valor a través de nuestro TFG. En esta, crearemos una experiencia similar a la vista en videojuegos comerciales como *Pokémon Go*. Este videojuego tendría la posibilidad de mejorarse y perdurar en el tiempo. Evitaríamos centrar las dinámicas de nuestro proyecto en torno a un modelo basado en micropagos, haciendo que el ejercicio sea la única forma de avanzar en el juego. Con estos puntos en cuenta, si conseguimos crear una experiencia entretenida, podemos lograr tener un videojuego de ejercicio atractivo para el público general, escalable y que fomente la actividad física. Con esto aportaremos una mayor visión a este campo con un juego de estudio con unas características parecidas a los títulos más comerciales.

Este videojuego podría ser usado en el futuro como una propuesta firme para luchar contra el sedentarismo y promover el ejercicio en los jugadores, de una forma divertida y social. A su vez, ayudaría a combatir el estigma que aún hoy existe de que los videojuegos están en contra de la actividad física y la socialización, contribuyendo a visibilizar que es una herramienta muy potente, como hemos visto, para abordar problemas como el sedentarismo en la sociedad actual.

Descripción del Trabajo

3.1. Diseño general de la propuesta

3.1.1. Premisa inicial: los critterons

La propuesta ha partido de una premisa muy sencilla: El juego, llamado *Critterons Capture*, consiste en ir capturando *critterons*, unas curiosas criaturas que el usuario debe alojar en su hotel, el cual podrá mejorar. Para poder obtener nuevos *critterons*, así como mejorarlos y ampliar las instalaciones del hotel donde se alojan, el jugador deberá desplazarse físicamente para acumular pasos. Éramos conscientes de que para que la experiencia resultara atractiva, los *critterons* debían ser el núcleo del juego, ya que son tanto el principal reclamo como el eje central de toda la experiencia. Estas son criaturas únicas, cada una con características propias y un diseño visual representativo. Además, cuentan con estadísticas de salud, ataque y defensa, así como ataques especiales únicos. Dichas estadísticas no son estáticas, sino que se ven afectadas por el nivel alcanzado por cada criatura, lo cual aporta al usuario una mayor sensación de progreso y vinculación con su colección.

3.1.2. Enfoque de la aplicación

En un primer momento, tuvimos que definir cómo queríamos enfocar la aplicación y qué nivel de interacción por parte del usuario deseábamos permitir. Para ello, analizamos varios enfoques posibles.

El primero, y nuestra propuesta inicial, consistía en construir una aplicación gamificada, donde la interacción del usuario fuera más guiada y limitada, similar a una propuesta como la que plantea *Duolingo*. Sin embargo, tras deliberar sobre esta opción, llegamos a la conclusión de que era necesario ofrecer mayor libertad al usuario.

Así, decidimos alejarnos de un formato más cercano al de aplicaciones tradicionales y educativas para proporcionar una experiencia más compleja que resultara atractiva tanto para el jugador casual como para el público general. La principal razón de esta decisión fue que, al no ofrecer una recompensa directa como podría ser la mejora del nivel de inglés del usuario, consideramos necesario enriquecer la experiencia lo suficiente para que el propio progreso dentro de la aplicación y la jugabilidad fueran recompensa suficiente.

Como segunda opción, valoramos desarrollar un juego más complejo, con una experiencia cercana a éxitos del sector como *Pokémon Go*. De este modo, el *gameplay* sería mucho más sólido y su atractivo, mayor. Además, conseguiríamos un valor añadido para los usuarios al eliminar pagos *in-game* y cualquier enfoque de monetización que entorpezca el progreso natural en el juego y la experiencia gratuita. Este formato nos pareció más compatible con nuestro objetivo. No obstante, detectamos una característica que hemos tenido que tener en cuenta: en estos juegos donde el *gameplay* es más profundo, si no se planifica adecuadamente, el desplazarse puede pasar a un segundo plano para el jugador, priorizando otras partes del juego como los gimnasios en *Pokémon Go*.

Para solventar este inconveniente, nos inspiramos en las aplicaciones deportivas, donde la interfaz es muy sencilla y no requiere atención constante, permitiendo al usuario centrarse en mantener un ritmo o alcanzar una distancia. Para aplicar este enfoque sin renunciar a la profundidad de la aplicación, decidimos mantener la complejidad en los combates y el hotel, elementos que consideramos suficientes para mantener el interés del jugador. Sin embargo, eliminamos distracciones de la pantalla principal mediante medidas como: prescindir de *critterons* dispersos por el mapa, emplear una interfaz en blanco y negro, y permitir que los combates se activen únicamente cuando el jugador lo desee. Con estos elementos, creemos que el usuario podrá centrarse en la actividad física sin perder la motivación para seguir utilizando la aplicación.

Tanto la dinámica final de la pantalla principal como la de los combates y el hotel pueden consultarse en el apartado de diseño.

3.1.3. Experiencia de juego

El juego está dividido en tres áreas principales:

- **Hotel:** donde el juego adopta una vista isométrica que permite al usuario desplazarse para explorar toda su extensión, similar a otros títulos de temática parecida como *Dragon City*. En este espacio se pueden observar todos los *critterons*, cómo interactúan entre sí, así como mejorar y expandir el hotel.
- **Combates:** donde el jugador puede enfrentarse a su *critteron* contra otros *critterons*, ya sea en solitario o junto a otros jugadores. Serán enfrentamientos

sencillos, presentados de forma similar a los combates de *Pokémon Go*. El jugador podrá utilizar tanto un ataque básico como alguno de los ataques especiales de su *critteron*, en un sistema por turnos que finalizará cuando todos los *critterons* de uno de los dos bandos se queden sin puntos de vida.

- **Desplazamiento:** donde el jugador pasa la mayor parte del tiempo. Este consistirá en una pantalla sencilla que muestra los pasos restantes hasta el siguiente encuentro con un *critteron*, así como las posibles apariciones, la vida actual y el progreso de nivel del *critteron* que lo acompaña, y los pasos necesarios para subir el nivel general del jugador.

El juego cuenta con diferentes dinámicas sencillas que se interconectan para hacer amigable la experiencia a todo tipo de jugadores: El jugador debe desplazarse para ir sumando pasos. Cada cierta cantidad de pasos se activará un combate contra otro *critteron* salvaje. Estos enfrentamientos permiten progresar tanto en el nivel de la criatura que lucha como en el nivel general del jugador. Además, en caso de ganar, se recibirán monedas. Dichos enfrentamientos desgastan al *critteron*, reduciendo su vida. Además, cada cierto número de enfrentamientos ganados, el jugador subirá de nivel general, lo que le permitirá obtener nuevos *critterons*, que serán invitados al hotel.

El jugador deberá seguir ganando combates no solo para subir de nivel a sus criaturas, sino también para conseguir monedas que le permitan ampliar el hotel. Esto dará acceso a nuevas habitaciones, que ofrecen mejoras como recuperación más rápida o zonas con nuevas interacciones entre las criaturas.

El jugador también podrá combatir contra otros jugadores o junto a ellos. Si se enfrenta a ellos, podrá obtener monedas extras; y si se une a ellos en sesiones de ejercicio cooperativo, accederá a combates más exigentes, pero con mejores recompensas.

Estos elementos sociales añaden variedad a las sesiones de juego y motivan aún más a los jugadores a subir de nivel y mejorar sus *critterons*.

Todas ellas requieren del desplazamiento del jugador para poder darse. Ya sean como consecuencia directa (moverse para llegar a las paradas y obtener sus monedas) o indirecta (realizar pasos para poder dar inicio a un combate). Al centrar todas las dinámicas del juego en torno al desplazamiento, usando este como el motor del juego, fomentamos la actividad física del usuario.

3.1.4. Ciclo de juego

Para que el juego funcione, requerimos un ciclo de juego robusto y se adapte a las necesidades de nuestra propuesta. Un ciclo de juego normal, desde que un jugador entra hasta que sale de la aplicación, es el siguiente:

En primer lugar, el jugador pasa por la pantalla donde inicia sesión o, si ya lo ha hecho anteriormente, avanza automáticamente. Este es el único punto que podemos considerar externo.

Tras entrar, el jugador se encuentra en la pantalla base, donde debe moverse mientras interactúa con paradas para obtener monedas, reduciendo el contador de pasos situado en la parte inferior y, así, poder dar inicio a un combate.

Una vez el contador de pasos llega a 0, el jugador puede iniciar un combate. Este se realiza automáticamente, pero el jugador tiene la opción de utilizar diferentes ataques, ya sean especiales o potenciadores del ataque básico, si quiere aumentar sus oportunidades de victoria.

Al completarse el combate, el jugador pasa a la pantalla de final de combate, donde se le mostrarán los resultados. En caso de haber ganado, recibirá recompensas en forma de experiencia y monedas. Si alcanza la experiencia suficiente, accederá a la pantalla de subida de nivel, donde podrá obtener un nuevo *critteron*, de un total de 12, o subir de nivel uno que ya posea, según la criatura que obtenga de la máquina.

Tras esto, el jugador será redirigido de nuevo a la pantalla base, para seguir caminando. En el juego encontramos dos pantallas principales, la base y la del hotel. Desde la primera se puede acceder tanto a la información del usuario como al hotel.

- En el Hotel, el jugador puede ver a sus *critterons* moverse e interactuar con ellos para consultar sus estadísticas. También es posible adquirir nuevas habitaciones de un total de 6 que otorgan mejoras, tales como ventajas en los combates, regeneración de vida, entre otros beneficios. Desde el Hotel se puede acceder también a la información del usuario.

- En la información de usuario, además de ver su perfil, cambiar el *critteron* actual y consultar estadísticas, el jugador puede acceder a la parte social de la aplicación. En esta sección es posible ver rankings globales o de amigos para comparar su progreso. También se puede gestionar la lista de amigos, añadiendo nuevos usuarios, que pueden ser seleccionados para combatir contra ellos o luchar junto a ellos.

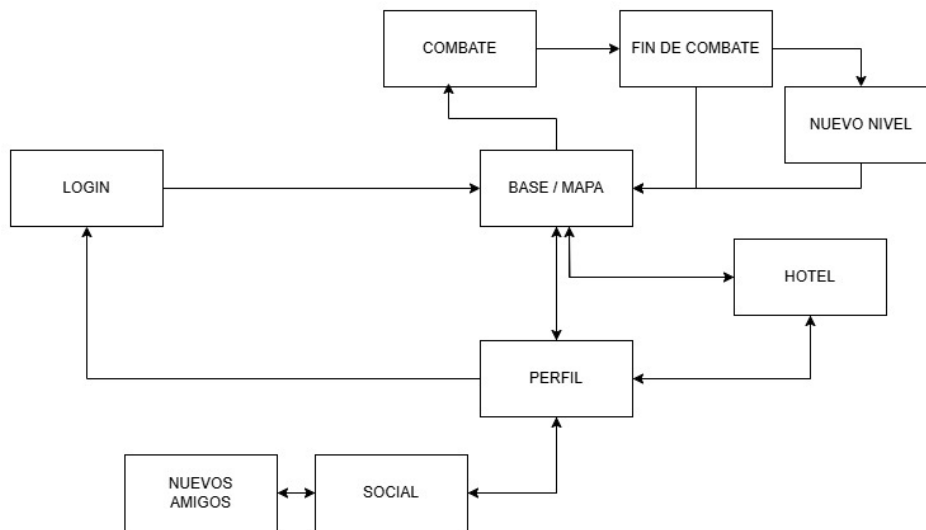


Figura 3.1: Navegación entre pantallas

El jugador navega entre todas estas pantallas, como podemos ver en la figura 3.1. Pudiendo variar el tipo de combate, utilizar diferentes *critterons* y expandir su hotel, mientras se mueve continuamente, recogiendo recompensas de paradas y participando en combates. Con estos progresos irá subiendo de nivel, adquiriendo nuevas criaturas para fortalecer su equipo y su hotel.

Así, se propone un ciclo de juego sencillo en su origen, pero con suficientes variantes para dotarlo de riqueza. El progreso rápido y sin necesidad de una toma de decisiones excesiva actúa como refuerzo para que el jugador continúe en el ciclo y prolongue sus sesiones de juego.

3.1.5. Progreso y retorno de jugador

El progreso es clave para mantener al usuario jugando y lograr que vuelva. Dada la naturaleza de nuestra propuesta, hemos optado por implementar un progreso sencillo, con una toma de decisiones muy reducida y continua.

Encontramos dos mecanismos con los que se avanza en el juego y se logra dicho progreso:

- **Las monedas:** se obtienen tanto al interactuar con paradas como al ganar combates. Sirven para ampliar el hotel, consiguiendo con esto mejoras pasivas, una mayor riqueza visual y la posibilidad de ver más *critterons* simultáneamente en el hotel. Al usarlas para comprar elementos, se convierten en una potente mecánica de progreso “tangible” para el jugador.
- **La experiencia:** se obtiene al completar combates y, una vez alcanzada cierta cantidad, permite que el jugador suba de nivel. Al subir de nivel, se consigue

un nuevo *critteron*; si esa especie ya la tenía, el *critteron* subirá de nivel; si no, se añadirá esta nueva especie al hotel y podrá combatir con ella.

Aunque estos dos elementos son los medios a través de los cuales el jugador avanza, los *critterons* son el verdadero atractivo y el núcleo que da valor a la propuesta. El poder conseguirlos todos, un tópico que tan bien funciona en otras franquicias como *Pokémon*, es el verdadero objetivo y la medida de progreso a la que más valor creemos que dará el jugador.

Al juntarlo con el hotel, que aunque sirva como soporte de este atractivo principal, es en sí mismo una medida de progreso llamativa, consideramos que contamos con un progreso claro, atractivo para el jugador y de fácil progresión, que encaja perfectamente con el tipo de juego que planteamos.

Otros elementos de progreso secundarios son:

- El nivel que alcanzan los *critterons*. El cual no tiene establecido un nivel máximo.
- El nivel del propio jugador. Al igual que el anterior no tiene definido un nivel máximo.
- Las monedas acumuladas en caso de haber obtenido todas las mejoras del hotel, las cuales al ya haber conseguido estas mejoras no tienen otro propósito.

Estos últimos están estrechamente ligados al retorno del jugador. Aunque conseguir todos los *critterons* ya nos parece una razón suficiente para fomentar la vuelta continua, el factor social es una riqueza añadida que consideramos puede potenciar aún más este retorno.

Que el jugador quiera superar a sus amigos y a otros jugadores en los rankings; que quiera tener más nivel que otros, *critterons* más poderosos, más monedas y, por supuesto, más *critterons* y un mejor hotel, lo valoramos como un punto clave que refuerza la adhesión al juego y fomenta el regreso de los jugadores a lo largo del tiempo.

3.1.6. Factor social

Existe una faceta multijugador que permite a los jugadores interactuar, enfrentarse y compararse con otros.

En primer lugar, el juego cuenta con un sistema de amigos, donde los jugadores pueden agregar a aquellos que les interesen y desbloquear con ellos rankings y dinámicas especiales.

Hay rankings, tanto de amigos como generales, para ver qué jugador es el que más distancia ha recorrido.

Dentro de las funciones con amigos, tenemos dos vertientes:

- **Faceta cooperativa:** se basa en realizar sesiones junto al *critteron* de otro jugador. En estas sesiones, los combates a los que se enfrenta el jugador son junto al *critteron* de su amigo y el suyo propio contra uno más poderoso. Estos combates son más desafiantes que los de un solo jugador, pero al ser superados, la cantidad de monedas obtenidas es mucho mayor.
- **Faceta competitiva:** permite al jugador enfrentarse al *critteron* de tus amigos. Si el jugador sale victorioso, obtendrá experiencia extra para su *critteron* y monedas, obteniendo experiencia y monedas en caso de triunfar en el combate.

Optamos por el momento por descartar una interacción directa entre dos amigos por la complejidad que esta puede suponer, pero creemos que poder interactuar con el *critteron* de tu amigo es suficiente y, además, muy representativo de la experiencia que queremos crear. Esto permite, por ejemplo, que te acompañe sin necesidad de tener que esperar a estar realmente juntos.

Con los rankings junto a los modos cooperativo y competitivo, creemos que el jugador encuentra un gran aliciente que ampliará en gran medida el tiempo de juego, tanto diario como con la aplicación en uso.

3.2. Entorno de desarrollo para el cliente

Para poder realizar un trabajo adecuado necesitamos un software que nos resulte cómodo de manejar y nos permita desarrollar toda la funcionalidad que requerimos para la aplicación. Con el objetivo de encontrar cuál es el más adecuado hemos buscado información de diferentes entornos de desarrollo para poder elegir el que más se adapte a las necesidades de la aplicación gamificada que queremos llevar a cabo.

Nuestros principales requerimientos son:

- Poder ser exportado para el sistema operativo de Android.
- Poder desarrollar juegos de carácter general, sin verse limitado a géneros o mecánicas concretas.
- Poder usar las funciones del dispositivo para importar la actividad física (podómetro, GPS, etc).
- Poder conectarse a internet para poder recolectar datos y permitir el factor social de la aplicación.

3.2.1. Candidatos potenciales

En primer lugar hemos obtenido una lista de las principales opciones disponibles y su propuesta de valor. En esta selección preliminar observamos que las opciones se separan en dos grandes grupos. Aquellas especializadas en videojuegos, que cuentan con mayor potencia pero una curva de entrada habitualmente mayor, y las de propósito general, algo más limitadas para crear videojuegos pero muy accesibles.

▪ Especializadas en videojuegos:

- **Unity:** motor de videojuegos popular con un editor potente que permite llevar a cabo juegos de toda índole y exportarlo rápidamente a todas las plataformas. Se centra tanto en juegos 2D como 3D y su comunidad es muy amplia.
- **Unreal Engine:** motor de videojuegos usado en la industria para proyectos de todas las categorías, desde indies a AAA. Cuenta con herramientas para generar gráficos, IA y juego online, entre otros, de primera categoría y posee una tienda de addons muy grande y de fácil acceso.
- **Godot:** motor de videojuegos open-source que ha adquirido popularidad en los últimos tiempos. En continuo desarrollo y con una comunidad cada vez más grande. Actualmente su punto fuerte se encuentra en los juegos 2D y cuenta con su propio lenguaje GDScript que lo hace más accesible.
- **Cocos2d-x:** motor de videojuegos open-source especializado en juegos 2D y programado en C++, lo que da un mejor rendimiento y un mayor control a la hora de desarrollar la plataforma. Usado sobre todo en el mercado móvil, para juegos casuales o híbridos.
- **LÖVE:** framework de desarrollo de videojuegos 2D, muy ligero y sencillo de usar, basado en el lenguaje de scripting Lua. Su principal fortaleza es lo sencillo que es de usar permitiendo crear juegos rápidamente, aunque se ve limitado cuando se intenta llevar a cabo proyectos más grandes.
- **Game Maker Studio:** motor de videojuegos muy accesible. Conocido por su sistema “drag and drop” que permite crear juegos, o partes de estos, sin tener que escribir código. Cuenta con un lenguaje de programación sencillo, GML, para proyectos que requieran más personalización. Usado sobre todo en proyectos de pequeña escala.
- **Corona:** framework de desarrollo de videojuegos basado en Lua. Es sencillo y fácil de dominar, por lo que permite crear juegos de manera rápida para móviles. Adecuado sólo para proyectos 2D de pequeña envergadura, cuenta a su vez con una comunidad activa.
- **Construct 3:** motor de desarrollo de videojuegos 2D, muy limitado para 3D, en navegador. Basado en “drag and drop”, permite programar en Java y de forma limitada, lo que hace la aplicación muy sencilla de aprender pero bastante limitada a la hora de realizar proyectos que requieran cierto grado de personalización.

- **LibGDX**: framework de desarrollo de videojuegos multiplataforma, open-source y flexible basado en Java. Permite realizar tanto juegos 2D como 3D. Permite crear aplicaciones donde se requiere un mayor control, personalización y rendimiento. Se usa sobre todo en desarrollo móvil y web y cuenta con una amplia comunidad.

▪ **De propósito general:**

- **React Native**: framework para desarrollo de aplicaciones móviles basado en JavaScript y React. Pensado para aplicaciones generalistas y UI, se puede usar para juegos simples. Permite desarrollo simultáneo para Android e iOS.
- **Flutter**: framework multiplataforma para aplicaciones con gran peso en la interfaz gráfica. Permite crear aplicaciones nativas para Android e iOS y se puede llegar a usar para juegos 2D.
- **Ionic**: framework multiplataforma basado en tecnología web (HTML, CSS y JavaScript). Enfocado en la creación de aplicaciones híbridas, se puede usar para crear juegos o aplicaciones gamificadas y tiene gran compatibilidad con otros frameworks.
- **NativeScript**: framework open-source y multiplataforma para crear aplicaciones nativas mediante JavaScript. Permite un acceso profundo tanto a las APIs nativas de Android como de iOS.

Ahora que ya hemos seleccionado los candidatos, hemos llevado a cabo un pequeño estudio mediante la documentación y foros de la comunidad de cada plataforma, para ver cuáles de las características que requerimos son implementables de manera sencilla. Cabe destacar que tuvimos que revisar algunos foros ya que opciones que a priori la plataforma señala como accesibles, requieren de una sobreingeniería que no era compatible con nuestro calendario. Algunas de las consultas fueron las limitaciones del multijugador en Construct 3, la dificultad para incluir soporte de red con LÖVE, guías para comenzar a desarrollar videojuegos con NativeScript, y la documentación oficial de Unreal Engine para exportar a Android.

Nombre	Compatible con las funciones API de Android	Amplia personalización	Soporte online en Android	Entorno adecuado para videojuegos
Unity	Presente	Presente	Presente	Presente
Unreal	Presente	Presente	Presente	Presente
Godot	Presente	Presente	Presente	Presente
Cocos2d	Presente	Presente	Presente	Presente
LOVE	Presente	Ausente	Ausente	Ausente
GameMaker Studio	Presente	Ausente	Ausente	Ausente
Corona	Presente	Ausente	Presente	Ausente
Construct 3	Presente	Ausente	Ausente	Ausente
LibGDX	Presente	Ausente	Ausente	Ausente
React Native	Presente	Presente	Presente	Presente
Flutter	Presente	Presente	Presente	Presente
Ionic	Ausente	Presente	Presente	Presente
NativeScript	Presente	Presente	Presente	Ausente

Tabla 3.1: Comparativa de motores y frameworks según nuestras necesidades

En la Tabla 3.1 en azul observamos aquellas características disponibles en ese motor, en rojo las no disponibles. Los principales motores de videojuegos como Unity o Unreal y frameworks de desarrollo generalista como React Native cumplen todas o la mayoría de características, mientras que son los motores de videojuegos más pequeños los que se encuentran más limitados.

3.2.2. Herramienta de desarrollo general o especializada en videojuegos

En primer lugar, analizamos si optábamos por un entorno de desarrollo de carácter general o por uno especializado. Ya que su propuesta de valor era a grosso modo similar dentro de los dos grandes grupos; y al tomar esta decisión, eliminaríamos un gran porcentaje de candidatos. Para esto extraímos y condensamos los puntos fuertes y débiles de cada una de las posibles soluciones (Tabla 3.2).

Tipo de entorno	Ventajas	Desventajas
Entornos generalistas	Mayor acceso a la API nativa de Android. Facilidades para trabajar con el componente online. Grandes comunidades con multitud de aplicaciones móviles de ejemplo.	Limitaciones al crear experiencias en tiempo real. Formas de interactividad reducidas. Soporte gráfico limitado, sin o con poco soporte para elementos 3D.
Motores de videojuegos	Enfocados en experiencias interactivas, con sistemas más adecuados para estas. Entornos con los que estamos más familiarizados. Plugins y addons ya adaptados al tipo de experiencia buscada.	Mayor complejidad para acceder a funciones nativas del dispositivo. Mayor dificultad para implementar funciones sociales por falta de addons específicos.

Tabla 3.2: Comparación entre entornos generalistas y motores de videojuegos

Con esta información y el estudio previo realizado, concluimos que la mejor opción para llevar a cabo nuestro proyecto del lado del cliente, dado que requerirá de gran interacción por parte del usuario, será la de realizarlo en un entorno especializado en videojuegos.

Pese a que el acceso a la API de Android con estos sea más complicado y partes del desarrollo como la faceta online y social puedan llevar una capa extra de trabajo, consideramos que las herramientas que se nos proporcionan, la estructura de trabajo de estos entornos, y la amplia cantidad de plugins de terceros pensados para videojuegos que pueden sernos de utilidad, hacen de esta opción la más adecuada para nosotros. Siendo el punto realmente determinante para seleccionarla, lo familiarizados que estamos con estos entornos, donde ya tenemos experiencia trabajando, sea en Unreal Engine o Unity. Esa experiencia nos será de utilidad para avanzar más ágilmente, evitar una gran barrera de entrada, independientemente del motor que seleccionemos, y, en general, permitirnos tener un desarrollo mucho más rápido.

El siguiente objetivo es seleccionar el motor con el que queremos llevar a cabo la aplicación.

3.2.3. Selección del entorno final donde desarrollaremos la aplicación

Una vez que habíamos acotado las opciones, eliminando los entornos generalistas, realizamos una selección posterior donde eliminamos aquellas herramientas especializadas en videojuegos pero que no eran compatibles con nuestros requisitos. Estos fueron LOVE, Corona y Construct 3, debido a su incompatibilidad con Android, ya sea a nivel general o con funcionalidades de estos , como la conexión de red.

Ahora sí, hemos podido centrarnos más en los restantes y analizar más detalladamente las ventajas y desventajas de cada uno (Tabla 3.3).

Motor	Ventajas	Desventajas
Unity	<p>Cuenta con un poderoso editor visual que permite trabajar mucho más rápido.</p> <p>Posee una gran comunidad con multitud de addons e información para el desarrollo móvil.</p> <p>Optimiza automáticamente los proyectos para obtener el mejor rendimiento posible.</p> <p>Cuenta con un gran soporte tanto 2D como 3D, permitiendo un acabado de primer nivel.</p> <p>Gran documentación.</p> <p>Soporte para multijugador.</p> <p>Tenemos gran experiencia con él.</p>	<p>Es un motor que requiere bastante potencia, pudiendo no ser compatible con algunos dispositivos más antiguos.</p> <p>Dependencia de plugins para multitud de funcionalidades nativas y relacionadas con entornos móviles.</p>
Unreal Engine	<p>Cuenta con un sistema de blueprints para poder desarrollar o expandir funcionalidades rápidamente.</p> <p>Hay una gran comunidad, tanto para crear plugins como para ayudar en los foros, que están a nuestra disposición.</p> <p>Gran documentación.</p>	<p>Pensado para proyectos de gran envergadura, puede añadir complejidad extra en pequeños proyectos.</p> <p>Se programa en C++, lo que aunque permite un mayor control, reduce ligeramente la velocidad del desarrollo.</p> <p>Es un motor que requiere bastante potencia, pudiendo no ser compatible con algunos dispositivos.</p>

Motor	Ventajas	Desventajas
	<p>Soporte para multijugador.</p> <p>Hemos trabajado con él, por lo que ya tenemos experiencia con este.</p>	<p>Centrado en el 3D, las funcionalidades para proyectos 2D se pueden sentir limitadas.</p>
Godot	<p>Motor open-source que permite trabajar a bajo nivel si se requieren funciones extra.</p> <p>Especializado en 2D, contiene multitud de herramientas orientadas a este tipo de gráficos.</p> <p>Su lenguaje GDScript es sencillo de usar, y también se puede programar en C#.</p> <p>Es un motor ligero que funciona en la gran mayoría de dispositivos.</p>	<p>Motor aún en crecimiento con multitud de funcionalidades que pueden ocasionar problemas o no funcionar como se espera.</p> <p>Sistema de plugins limitado.</p> <p>Comunidad para el desarrollo móvil más reducida.</p>
Cocos2d	<p>Motor open-source que permite trabajar a bajo nivel si se requieren funciones extra.</p> <p>Es un motor ligero y muy optimizado, orientado a desarrollo móvil, que podrá ser usado en la gran mayoría de dispositivos.</p> <p>Especializado en 2D, contiene multitud de herramientas orientadas a este tipo de gráficos.</p> <p>Comunidad activa con multitud de foros disponibles y buena documentación.</p> <p>Además de C++, soporta lenguajes más cómodos de trabajar como C# y JavaScript.</p>	<p>Sistema de plugins limitado.</p> <p>El desarrollo móvil requiere optimización manual a diferencia de otros motores más automatizados.</p> <p>Aunque permite usar Lua o Java, su soporte principal es en C++, lo que puede reducir la velocidad de desarrollo.</p>
GameMaker	<p>Interfaz visual potente y amigable que permite un desarrollo más rápido.</p>	<p>Problemas de rendimiento si el juego tiene multitud de elementos.</p>

Motor	Ventajas	Desventajas
	<p>Especializado en 2D, contiene multitud de herramientas orientadas a este tipo de gráficos.</p> <p>Comunidad activa con multitud de foros disponibles y buena documentación.</p> <p>Soporte para multijugador.</p>	<p>Sistema de extensiones limitado.</p> <p>El desarrollo móvil requiere optimización manual a diferencia de otros motores más automatizados.</p>
LibGDX	<p>Motor open-source que permite trabajar a bajo nivel si se requieren funciones extra.</p> <p>Se programa en Java, lenguaje que conocemos.</p> <p>Buen rendimiento, permitiendo alcanzar multitud de dispositivos móviles.</p> <p>Comunidad activa con multitud de información y soporte disponible.</p> <p>Permite integrar una interfaz fácilmente, agilizando el desarrollo.</p> <p>Soporte para multijugador.</p>	<p>Curva de aprendizaje al ser un framework de más bajo nivel.</p> <p>Sin editor visual, lo que reduce en gran medida la velocidad de trabajo.</p> <p>El uso de funcionalidades nativas puede requerir trabajar con código a bajo nivel.</p>

Tabla 3.3: Resumen de ventajas y desventajas de los diferentes motores

Tras sopesar las diferentes opciones, llegamos a varias conclusiones:

Debíamos trabajar con un motor que contara con una interfaz gráfica, fuera nativa o de fácil integración, si queríamos que el desarrollo fuera lo más ágil posible. Sin ella, el realizar cambios en la interfaz y modificar los elementos visuales de la aplicación se habría vuelto más lento.

Queríamos trabajar en un lenguaje que nos resultara sencillo y amigable. Aunque teníamos bastante experiencia con C++, tener que trabajar con memoria a más bajo nivel o complicaciones similares nos habría hecho avanzar más lento y nos podría haber dado complicaciones conforme se expandiera la aplicación.

Cuanto más grande fuera la comunidad, tanto en la creación de plugins externos como en su ayuda en discusiones y foros, más sencillo sería nuestro trabajo. Esto nos permitiría solventar los problemas que ocurrieran durante el desarrollo mucho más rápido y no complicar más una agenda que tendría que mantenerse lo más constante

posible para poder llevar a cabo las pruebas con usuarios a tiempo.

Debía tener un soporte nativo para implementar funciones multijugador, aunque tuviéramos que expandirlo nosotros con plugins externos o de forma manual. Ya que si hubiéramos tenido que implementarlo nosotros desde cero, el coste de desarrollo del cliente habría sido mucho más alto.

Por último, aunque no era estrictamente necesario, dado que el proyecto a desarrollar era bastante grande y tenía multitud de facetas, preferíamos tener experiencia en el motor para poder comenzar a desarrollar desde las primeras fases sin tener que pasar por un período de aprendizaje.

Con toda la información recopilada y los puntos más importantes claros, decidimos usar Unity como nuestro entorno de desarrollo.

Es un potente motor para videojuegos con multitud de herramientas para el desarrollo móvil. Contaba con funciones multijugador y sociales fácilmente implementables. Tenía una gran comunidad y multitud de addons que podíamos incorporar rápidamente a nuestro proyecto. Su interfaz era muy completa y permitía prototipar y modificar la aplicación rápidamente. Y, añadido a todo lo anterior, ya contábamos con amplia experiencia trabajando con él, pudiendo saltar la barrera de entrada y comenzar de pleno con el desarrollo desde el principio.

3.3. Desarrollo del cliente en Unity

Al ser un proyecto bastante complejo y que ha requerido de trabajo en diferentes áreas optamos por trabajar de manera iterativa. Así buscamos ir logrando hitos dentro del proyecto que dieran como resultado versiones con funcionalidad reducida de la aplicación pero funcionales. Que nos han dado la capacidad de probar diferentes aspectos de la aplicación, comprobar que la integración de la API es correcta y corregir aspectos de la jugabilidad y experiencia de juego.

3.3.1. Hitos durante el desarrollo

Al llevar a cabo un desarrollo iterativo, nos propusimos múltiples hitos para la aplicación que hemos ido alcanzando. En cada uno de estos hitos hemos ampliado la funcionalidad de la aplicación, manteniendo en todos ellos un prototipo funcional con el que poder ir probando otros elementos del proyecto como la API. Estos hitos son propios del cliente y se han llevado a cabo durante las etapas del desarrollo: desarrollo del videojuego, pruebas con usuarios y análisis de datos.

- **Primer hito, hotel funcional:** como la mayoría de los datos con los que trabaja el juego tiene presencia en el hotel, concluimos que la primera parte sobre

la que debíamos de trabajar era esta; para poder tener un entorno real sobre el que probar la API. Además, dicha escena es la que más peso de desarrollo ha tenido. Al concluir este hito habíamos conseguido un hotel funcional, con los assets finales tanto de *critterons* como de habitaciones, capaz de comunicarse con la API; aunque aún sin posibilidad de interacción por parte del usuario.

- **Segundo hito, combate y funciones sociales:** otros dos elementos claves de la experiencia son los combates, necesarios para subir de nivel y un punto clave en el ciclo de juego, como el factor social, que requiere del otro gran volumen de llamadas a la API. Su importancia en la aplicación, así como su complejidad, hizo que fueran el siguiente grupo de elementos a desarrollar. Una vez completo habíamos conseguido un combate funcional sin errores mayores y una estructura de pantallas que permitían realizar todas las funciones sociales a las que tiene acceso el usuario.
- **Tercer hito, mejora del hotel y el combate:** primera fase de desarrollo donde el objetivo no era crear nuevas escenas sino mejorar y ampliar las actuales. Se pulió el combate para evitar errores menores y mejorar la experiencia, y se añadió interacción en el hotel para que el usuario pudiera interactuar con *critterons* y realizar la compra de habitaciones. Primera versión con exportación a dispositivos Android.
- **Cuarto hito, geoposicionamiento, podómetro, pantalla principal y subida de nivel:** en este se añadió la pantalla principal, así como el podómetro y el geoposicionamiento con conexión directa con esta escena. Junto a la adición de pantallas de subida de nivel y fin de combate. En este hito fue la primera vez que se alcanzó una versión del juego con el ciclo de juego completo funcional.
- **Quinto hito, ampliación de la experiencia de juego:** se expandió el juego para agregar multitud de elementos que enriquecen la experiencia de juego. Algunos fueron los combates con amigos o en equipo, la carga de paradas dinámicas desde el servidor, la muestra de perfiles de otros jugadores, entre otros.
- **Sexto hito, arreglo de errores y mejora de la experiencia de usuario:** se arreglaron multitud de errores menores que entorpecían el uso de la aplicación y se rediseñaron pantallas para mejorar la experiencia del usuario con estas, buscando facilitar la navegación o estructurar mejor la información mostrada.

Gracias a este enfoque hemos tenido versiones funcionales durante todo el desarrollo que nos han permitido seguir trabajando en paralelo con otras partes del desarrollo mientras mejorábamos el juego.

3.3.2. Estructura del proyecto

La estructura del proyecto sigue un formato estándar que nos ha permitido trabajar fácilmente. Todo el trabajo que hemos realizado en la aplicación lo hemos mantenido dentro de la carpeta de *Assets*. Dentro de esta encontramos multitud de carpetas, autoexplicativas en el nombre, señalando las más importantes:

- **Assets:** contiene los *assets* propios, estos incluyen los elementos visuales para la *UI*, los 3D pertenecientes tanto a los *critterons* como a las habitaciones, los sonidos que hemos utilizado en el juego y el icono de la aplicación.
- **Prefabs:** contiene los diferentes *prefabs*, plantillas de objetos que podemos usar múltiples veces una vez definidas, que reutilizamos en múltiples escenas.
- **Scripts:** contiene los *scripts* principales del juego que dotan de funcionalidad al juego, separados a su vez en carpetas por funcionalidad.
- **Scenes:** contiene todas las escenas en las que se divide el juego, tanto las finales como las de prueba, separadas, que usamos para la maquetación y fases iniciales del desarrollo.
- **StepCounter y Geoposicionamiento:** contienen toda la funcionalidad de estos elementos, separados del resto de la solución para mantener ambos módulos aislados.

3.3.3. Principales clases, modularidad y escenas

El juego es un proyecto bastante complejo que nos ha llevado a desarrollar multitud de *scripts* (archivos ejecutables) y, por tanto, de clases, para dotar de funcionalidad a toda la estructura. Tanto de cara al usuario como por debajo, en el backend de la aplicación. Múltiples clases más realizan tareas adicionales y hacen de auxiliares a las principales clases del proyecto, que son:

- **HotelManager, HotelInput y NavigationControl**, que se encargan de la gestión del hotel, la interacción del usuario con el hotel a nivel visual y lógico, y el movimiento de los *critterons* respectivamente.
- **CombatManager y CombatUI**, que se encargan de la gestión general del combate y de su representación en la interfaz.
- **AudioManager**, que controla el audio en todo el juego.
- **Machine**, que gestiona la escena de subida de nivel.
- **SelectPanel y ButtonActions**, *scripts* generalistas que nos han permitido desarrollar menús animados mucho más rápido, encapsulando las diferentes acciones al interactuar con ellos.

- **Login**, que gestiona la escena de *login* y es la encargada de permitir el acceso del jugador al resto del juego.

Hemos dejado de lado las pertenecientes al servidor, *geoposicionamiento* y *podómetro*, ya que los consideramos módulos separados que requieren de su propia sección más adelante.

Todas estas clases y su división clara de tareas son posibles gracias a la modularidad que posee el proyecto. Hemos optado por separar las diferentes secciones del juego todo lo posible para reducir los errores potenciales y simplificar su gestión. Para hacer esto correctamente hemos usado las escenas, que permiten dividir un juego en diferentes secciones sin relación directa, una herramienta nativa de *Unity* muy potente y fácil de usar.

Encontramos diferentes módulos o secciones que pueden estar formados por una o más escenas:

- **Hotel**, donde el usuario puede ver los *critterons* y comprar mejoras. Esta sección está compuesta por una sola escena: Hotel (Figura 3.2).



Figura 3.2: Escena del hotel. Izquierda vista general, derecha seleccionado un critteron

En la imagen observamos la escena del hotel, donde encontramos *critterons* en diferentes habitaciones. En la parte de arriba, el usuario puede ver tanto su nivel como su nombre y monedas disponibles. Puede interactuar tanto con los *critterons* como las habitaciones para desplegar cuadros de diálogo que le den información del objeto, como se muestra en la captura de la derecha.

- **Combate**, donde el usuario se enfrenta a otros *Critterons* y sube de nivel. Está compuesto por tres escenas: Combat, EndCombat y NewLevel. Al ser el

combate la única forma de subir de nivel, hemos optado por combinar todo este recorrido como un mismo módulo (Figura 3.3).



Figura 3.3: Escena del Combate

En la imagen observamos la escena de combate. Más cerca de la cámara se encuentra el *critteron* del jugador, mientras que al fondo está el rival. En la parte de arriba se indica qué tipo de batalla es, en este caso, contra uno salvaje. Los cuadros de información muestran el tipo, el nivel y la vida del *critteron*; a la izquierda, el del rival, y a la derecha, el del jugador. En la parte de abajo se muestran tres botones que permiten al jugador seleccionar el siguiente ataque de su critteron.

- **Social**, donde el usuario puede interactuar con otros usuarios, ver *rankings* y activar actividades como combatir contra otros jugadores o luchar a su lado. Cuenta con dos escenas: Social y AddFriend. Optamos por reducir al máximo el número de escenas para facilitar la navegación (Figura 3.4).



Figura 3.4: Pantallas en social. Izquierda lista de amigos, derecha ranking de amigos

En la imagen observamos dos pantallas dentro de la parte social. La de la izquierda muestra la lista de amigos y permite al usuario acceder a los rankings con los botones de abajo. La de la derecha muestra el ranking de amigos. En ambas, tenemos botones en la parte superior que nos permiten agregar amigos e ir al hotel.

- **Estadísticas de usuario**, donde el usuario puede observar el número de pasos y ver datos sobre el progreso del juego. Además puede modificar el *critteron* con que va a combatir. También es el punto principal para viajar a otras escenas (Figura 3.5).

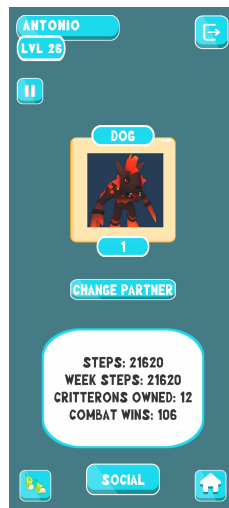


Figura 3.5: Escena del estadísticas de usuario

En la imagen observamos la pantalla de perfil del usuario. En la parte superior, el nombre, el nivel y un botón para cerrar sesión. En la parte inferior, botones para dirigirnos a diferentes escenas de la aplicación. En el centro, encontramos el *critteron* equipado, el botón para cambiarlo y las estadísticas generales del usuario.

- **Main**, que incluye tanto la pantalla principal que el usuario ve al moverse, desde la que puede interactuar con *paradas* e iniciar combates, como la pantalla donde ve su información actual. Consta de dos escenas: Principal y UserInfo (Figura 3.6).

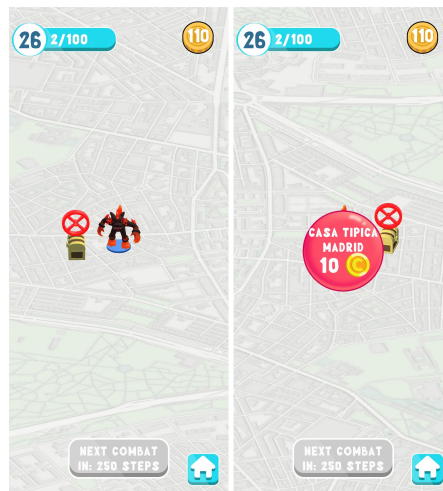


Figura 3.6: Escena principal. Izquierda vista general, derecha seleccionada una parada

En la imagen observamos dos capturas de la escena base. A la izquierda, la vista general mientras se desplaza y, a la derecha, el *popup* cuando se interactúa con una parada. En la parte superior encontramos información general del usuario y, en la inferior, el botón para dar inicio al combate y dirigirse al hotel.

- **Login**, desde donde el usuario entra al juego, es la más aislada de todas. Solo cuenta con la escena de Login y en ella puede iniciar sesión o crear un nuevo usuario (Figura 3.7).

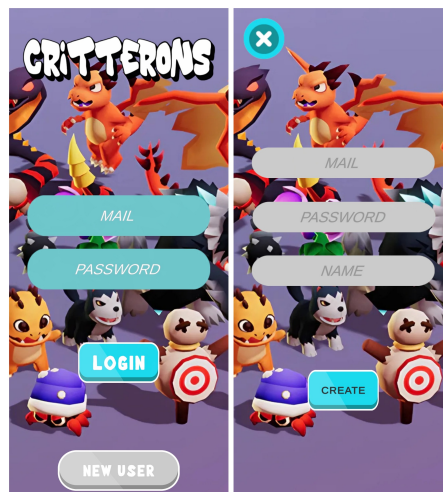


Figura 3.7: Escena de login. Izquierda inicio de sesión, derecha creación de una cuenta.

En la imagen observamos la pantalla de inicio de sesión. A la izquierda, la vista inicial, donde podemos introducir los datos para iniciar sesión o crear un nuevo usuario con el botón de abajo; y a la derecha, la vista para crear un nuevo usuario.

3.3.4. Assets y addons usados para el proyecto

Realizar todo el trabajo por nuestra cuenta sin usar paquetes externos hubiera supuesto un volumen de trabajo mucho mayor del que podíamos permitirnos, por lo que hemos utilizado varios paquetes externos para agilizar el desarrollo de la aplicación. Podemos dividirlos en dos grupos.

El primero son aquellos que nos ayudan a extender la funcionalidad de la aplicación. En este encontramos:

- **DOTween:** para crear animaciones y movimientos dinámicos de manera mucho más sencilla.
- **Newtonsoft Json:** para la gestión de archivos *json*, que nos sirven tanto en la configuración como para trabajar con la comunicación con la API.

Por otro lado, tenemos aquellos relacionados con el apartado visual. Han sido claves en el desarrollo, pues esta es la parte donde más ayuda externa necesitábamos, ya que para lograr un trabajo aceptable habríamos tenido que dedicarle mucho tiempo. Gracias a estos *assets*, solo hemos tenido que trabajar en su integración y adaptar algún elemento. Estos son:

- **Sweet Land GUI y Mega Hyper Casual GUI Pack:** de los que hemos obtenido los *sprites* con los que hemos montado la interfaz de usuario.
- **Cartoon FX Remaster:** que nos proporciona sistemas de partículas, elementos que nos generan efectos visuales vistosos, permitiéndonos dar mayor *feedback* y riqueza visual a la experiencia.
- **Cozy Cartoon Rooms Interior:** que contiene los modelos para las múltiples habitaciones que conforman el hotel; tanto las propias habitaciones como los *props* (muebles, decoración, etc.), así como el modelo de los cofres para el mapa.
- **Monsters Ultimate Pack Cute Series:** paquete que incluye modelos y animaciones de criaturas, clave para poder crear los *critterons* que protagonizan el juego, a raíz de estos modelos visuales.

Los sonidos los hemos obtenido de recursos de licencia abierta en YouTube. Otros *assets* visuales, como imágenes para completar la interfaz, los hemos obtenido de librerías online de recursos de uso abierto, como Flaticon, que permiten usarlos sin restricciones, ya sea de manera general o con suscripciones.

Los paquetes que hemos usado han sido gratuitos, o paquetes que ya poseíamos antes de comenzar con el desarrollo del TFG.

3.3.5. Geoposicionamiento y su integración

Siguiendo el ejemplo de juegos como *Pokémon Go*, hemos querido proporcionar al jugador recompensas por desplazarse en el espacio y no solo dar pasos alrededor de un solo punto; evitando así la repetición y dotando al juego de una experiencia más rica. Para esto, el juego cuenta con un mapa que muestra tu posición actual en el mundo y paradas con las que puedes interactuar para obtener monedas si te encuentras cerca de ellas.

El principal desafío técnico para lograr esto es disponer de un mapa actualizado con tu posición GPS y que incluya las paradas sobre este en la posición correcta, además de contar con un diseño adecuado para que no moleste al jugador ni lo distraiga en exceso.

Soluciones como *OpenStreetMaps* nos dan opciones para construir el mapa desde bajo nivel. Pero dado que esto requiere un gran volumen de trabajo, hemos optado por usar add-ons de terceros y modificarlos según nuestras necesidades, para poder tener un resultado aceptable sin dedicarle tiempo excesivo.

El add-on utilizado es Online Maps de *Infinity Code*, que podemos encontrar en la Unity Asset Store, el cual ya poseíamos. Esta es una solución que permite integrar mapas y funciones de geoposicionamiento de manera sencilla, con gran flexibilidad y multitud de servicios para elegir si se quiere profundizar en las opciones que ofrece.

Su integración fue sencilla gracias a la extensa documentación y los ejemplos que vienen con el propio *add-on* al importarlo en Unity. Hemos tenido que realizar modificaciones en el código para adaptarlo a nuestras necesidades, como la modificación de controles. La configuración que hemos establecido para el mapa, ya que da opciones de multitud de fuentes y formatos, es provista por *CartoDB* y del tipo *Positron*, dando una estética en blanco y negro que resalta las paradas y no acapara la atención del jugador. Las paradas y el jugador, representado por su critteron en el centro de la escena, destacan sobre este mapa simple y con una interfaz minimalista (Figura 3.8).

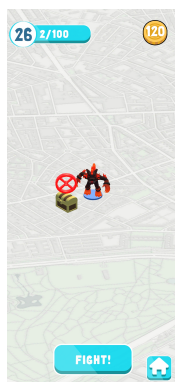


Figura 3.8: Ejemplo mapa

A su vez, hemos tenido que crear diferentes clases para gestionar y encapsular la interacción entre el juego y el contenido que tenemos que mostrar sobre el mapa, con la propia solución de `Online Maps`.

Por un lado, la clase `MarkControl` se encarga de obtener y construir la información de las paradas que son pasadas a `OnlineMaps` para integrarlas en el mapa, así como controlar las interacciones de las paradas, el tiempo entre ellas y dar al jugador las recompensas una vez completadas.

Las clases que dotan de funcionalidad a las paradas son `InteractMarkBehaviour` y `Mark3D`. La primera se encarga de la parte visual y el *feedback* de las paradas, y la segunda de lanzar y mostrar visualmente la interacción con estas.

Con todo esto, tenemos un mapa y posición global que se actualizan constantemente y permiten al jugador disfrutar de una experiencia más completa, con mayor *feedback* y que le da alicientes para desplazarse más allá de sus lugares típicos mientras hace ejercicio.

3.3.6. Integración del podómetro

Hemos tenido claro desde el principio que el objetivo del juego es fomentar el movimiento y el ejercicio, por esto la principal medida de progreso es la de la distancia desplazada. Aunque el geoposicionamiento era una opción para medir este movimiento, su precisión podría verse reducida y repercutir en el progreso del jugador.

Es por esto que hemos optado por usar un sistema auxiliar y disponible en la gran mayoría de dispositivos Android, como es el podómetro, para controlar cuántos pasos realiza el jugador, y con ello cuánto ejercicio está llevando a cabo y poder recompensarlo en consecuencia.

Para implementarlo, buscamos inicialmente *add-ons* disponibles en la Unity Asset Store o compatibles con el motor, pero no encontramos opciones gratuitas, por lo que hemos tenido que optar por desarrollarlo manualmente.

Hemos desarrollado una versión inicial donde, mediante el acelerómetro y con un proceso posterior, obtenemos un aproximado de los pasos realizados. Pero no era muy precisa y era común superar por mucho el umbral de paso que considerábamos como válido. Nuestro siguiente acercamiento fue usar scripts y soluciones creadas por la comunidad, pero las opciones eran limitadas y no llegaron a convencernos.

Es por esto que optamos finalmente por incorporar el nuevo sistema de *input* de Unity en el proyecto, con el cual hemos tenido acceso al sensor del podómetro nativo de Android, pudiendo ceder el conteo de pasos al propio sistema. Sobre este enfoque creamos la clase `StepCounterV2`. Clase independiente que gestiona todo lo relacionado con el seguimiento de pasos del dispositivo y le permite al juego usar

estos datos para disparar los combates y, por tanto, permitir al jugador avanzar. Sus principales funciones son:

- Inicializar el sensor y todo lo necesario para poder realizar el seguimiento mientras la aplicación está activa, ya sea en primer o segundo plano.
- Solicitar los permisos necesarios para poder acceder al sensor del dispositivo.
- Encapsular la gestión del sensor y limitar el número de pasos máximos para evitar malfuncionamiento.
- Actualizar el registro de pasos para que sea utilizable por otras partes del juego.

Con esta implementación hemos logrado tener un módulo independiente, sencillo, que encapsula la gestión de los pasos que realiza el jugador, obteniéndose directamente del sensor nativo del dispositivo y, por tanto, logrando una precisión mucho mayor que con otras soluciones.

3.3.7. Integración de la API en Unity y optimización de recursos

Para integrar la API en nuestro juego en Unity hemos tenido que crear toda la estructura del cliente dentro de nuestro proyecto. Hemos decidido que los principales valores a tener en cuenta a la hora de desarrollarla sean:

- **Independiente**, para evitar la mayor cantidad de fallos potenciales y reducir su mantenimiento cuando realizáramos cambios sobre el juego.
- **Un único punto de entrada/salida**, para facilitar la depuración y concentrar las responsabilidades.
- **Robusta**, capaz de controlar situaciones inesperadas como fallos de conexión o tiempos de respuesta largos.
- **Modular**, separando los diferentes tipos de información en diversas clases para facilitar la gestión y la ampliación de sus capacidades.

La clase principal, y que realiza las conexiones propiamente dichas con la API, es `ServerConnection`. Por ella pasan todas las peticiones y respuestas, y toda la configuración relacionada con estas; es la columna vertebral de este módulo del proyecto. Su método `SendRequest` es el encargado de realizar estas peticiones, contando con multitud de métodos que encapsulan este para solicitar y volcar los diferentes tipos de datos que el juego requiera. Otro punto clave de su funcionamiento es que, dado

que la comunicación con la API no se realiza al instante, la gran mayoría de sus funciones públicas que son usadas por las otras clases son asíncronas, operaciones que no bloquean el hilo principal. Dentro de estas encontramos dos enfoques:

- **Corrutinas:** propias de Unity y fáciles de implementar. Las hemos usado para subir datos, cuando no se requiere esperar a que se complete dicha tarea para continuar con el flujo de trabajo.
- **Tasks:** más modernas y propias de C#, algo más complejas en su implementación, permiten controlar el flujo de ejecución esperando a que se completen mediante *await* (funcionalidad que permite suspender la ejecución hasta que la tarea asíncrona finalice). Las hemos usado cuando solicitamos datos, y necesitamos esperar a que estén disponibles para continuar.

Esta clase, al encargarse de la comunicación con la API, es también la que gestiona la configuración de red, mediante la lectura de un archivo de configuración JSON.

Las siguientes clases relevantes son `RequestGameInfo`, que es la encargada de obtener los datos generales del juego, y `RequestUserInfo` y `RequestUserSocialInfo`, que se encargan de obtener los datos particulares del usuario o de otros jugadores. Estas clases actúan como *wrappers*, encapsulando las llamadas a `ServerConnection` y devolviendo de manera mucho más clara y ordenada toda la información que el jugador necesita durante la sesión de juego.

Todas estas clases siguen a su vez el patrón singleton, donde solo puede haber una instancia de la misma y se accede a esta mediante un método estático. Con este enfoque hemos buscado evitar llamadas extras y posible duplicidad o corrupción en los datos recibidos y enviados.

La información que se necesita durante una sesión incluye elementos tan variados como el nivel actual, experiencia obtenida por combate o los nombres de sus amigos. Para poder gestionarla correctamente, trabajar con ella y almacenarla durante la sesión, hemos creado multitud de interfaces que son las que las clases externas a este módulo usarán para acceder a los datos obtenidos; algunas de ellas son `I_Critteron` e `I_User`.

Una clase externa a este flujo de trabajo es la de `Login`. Esta se encarga de la creación de nuevos usuarios, la gestión y creación de tokens, el inicio de sesión mediante estos y la gestión de la propia escena donde se encuentra. Se comunica directamente con el `ServerConnection`.

3.3.8. Integración de Xasu

Para recolectar los eventos que ocurren durante el juego y poder obtener un mapa del recorrido del usuario dentro del videojuego, hemos usado *Xasu*, un *asset* de

tracking de datos desarrollado por el grupo e-UCM de fácil integración en Unity. Con él hemos registrado multitud de eventos que ocurren dentro del juego conforme el usuario interactúa con él para obtener un análisis mucho más detallado de cómo el jugador ha usado la aplicación. Esto ha sido clave a la hora de poder sacar conclusiones y ver si realmente el ciclo de juego funciona.

Algunas de las acciones que se registran como eventos son cuando:

- El jugador gana un combate.
- El jugador cambia de escena.
- El jugador interactúa con una parada.
- El jugador compra una habitación.
- El jugador obtiene un nuevo critteron.

Como se puede observar, gracias a la flexibilidad de la herramienta se pueden registrar todo tipo de eventos. Para esto se combinan diferentes verbos (*completed*, *unlocked*, etc.) con un mensaje y parámetros opcionales que se le pueden enviar. Juntando todo esto nos da un completo registro del que podemos sacar multitud de información y una imagen clara del recorrido del jugador. Estos registros son almacenados con el formato estándar de la experiencia API (xAPI), un formato que almacena experiencias a través de declaraciones "sujeto-verbo-objeto". Lo que nos permite poder exportarlos y trabajar con ellos de manera más sencilla que si siguiéramos un formato propio.

Debido a que hemos integrado esta herramienta en un momento muy tardío, cerca de las pruebas, no nos dio tiempo a configurar la conexión con el servidor del grupo e-UCM, por lo que hemos guardado los datos en local en el dispositivo del usuario. La estrategia alternativa fue implementar un mecanismo para que este pudiera descargarlos y, una vez completadas las pruebas, enviarlos. Aun con alguna pérdida de datos por error humano del usuario, hemos logrado recoger la gran mayoría de los datos para su posterior análisis.

La interacción de todo el ecosistema con la herramienta de Xasu se realiza mediante el script que hace de *wrapper*, una encapsulación de la lógica de Xasu, `XasuControl`; que hace de puente entre ambas partes.

3.3.9. Pulsera EmotiBit

Durante la primera etapa del diseño del TFG, se exploró utilizar dispositivos externos con el objetivo de explorar su integración en la aplicación para ofrecer funcionalidades adicionales exclusivas a los usuarios que la poseyeran. Con esto se buscaba ampliar la experiencia de juego, ofreciendo una capa extra para dicho grupo.

Tras sopesar diferentes opciones, nos decantamos por la solución de *EmotiBit*. Una pulsera con software propio, exportación de datos y documentación; a la que teníamos acceso.

Las pulseras *EmotiBit* son dispositivos diseñados para medir diversos parámetros corporales, incluyendo, pero no limitándose a: sudoración, pulso y tensión arterial. Gracias a su amplia capacidad de medición, nos planteamos utilizarla para ampliar el control sobre la actividad física del usuario, ofreciendo recompensas y funciones exclusivas que incentivaran un mayor esfuerzo físico más allá del simple desplazamiento.

Llevamos a cabo distintas pruebas con la pulsera para comprender su funcionamiento y evaluar su grado de integración con la aplicación. Durante estas pruebas logramos:

- Preparar el entorno de desarrollo.
- Conectar el dispositivo correctamente.
- Obtener varias mediciones en tiempo real.
- Descargar algunos datos en crudo.

Tras estas pruebas, llegamos a las siguientes conclusiones:

1. La pulsera no puede conectarse al dispositivo móvil vía *Bluetooth* ni *Wi-Fi*, lo que hace inviable una conexión directa con el cliente.
2. En nuestro caso, el dispositivo tenía una batería muy limitada, insuficiente para realizar actividades físicas extensas con la pulsera conectada.
3. El software proporcionado por los desarrolladores únicamente permite obtener datos en crudo, por lo que cualquier sistema de análisis o recompensas debía ser desarrollado desde cero.
4. La principal forma de recoger los datos era mediante una conexión física por puerto USB, lo que hacía inviable su uso por parte del usuario final.
5. La documentación disponible estaba en gran parte desactualizada, y funcionalidades como una app de Android para sincronizar datos habían sido descontinuadas.
6. La comunidad de usuarios presentaba los mismos problemas que nosotros y no parecía haber soluciones disponibles.

Cabe destacar que el código proporcionado por la empresa es de código abierto, por lo que hubiera sido posible ampliar las prestaciones que echamos en falta. Sin

embargo, llevar a cabo estas mejoras requería un volumen de trabajo superior al que consideramos razonable para esta parte del proyecto.

Por todos estos motivos, decidimos descartar el uso de la pulsera *EmotiBit*, a pesar de su gran potencial, y centrarnos exclusivamente en los pasos registrados por el dispositivo móvil como métrica principal.

3.4. Desarrollo del servidor

Dado que el proyecto consiste en un videojuego en línea, uno de los mayores retos es definir y estructurar adecuadamente los datos que el sistema debe gestionar. Esto implica tanto el diseño del modelo de datos como la elección de la tecnología adecuada para su almacenamiento y posterior acceso. Además de la información local asociada a cada usuario, es necesario desarrollar una parte del sistema que actúe como servidor, permitiendo el intercambio de información entre jugadores.

Asimismo, el servidor debe ser capaz de gestionar y actualizar datos, facilitando la introducción de contenido sin necesidad de modificar directamente el cliente. Esto permite añadir nuevos *critterons*, ampliar o modificar partes del hotel o ajustar estadísticas del juego, entre otros.

3.4.1. Estructura de los datos

Antes de definir cómo se almacenan y gestionan los datos, es fundamental entender qué tipo de información necesita el juego. Los datos que se guardan están relacionados tanto con los propios jugadores como con los elementos del juego, incluyendo las características de los *critterons*, las habitaciones y otros componentes.

En la Figura 3.9 se puede observar cómo las diferentes colecciones están interrelacionadas entre sí. Estas colecciones son:

- **User:** Representa a cada uno de los jugadores de la aplicación. Contiene información principal del jugador, como sus características actuales, estadísticas y el progreso alcanzado dentro del juego.
- **Critterons:** Como se comentó anteriormente, los *critterons* son el elemento central del juego. En esta colección se almacenan sus estadísticas, niveles para desbloquearlos, habilidades y otros atributos relevantes.
- **Room:** Representa cada una de las habitaciones presentes en el juego incluyendo los tipos o precios de estas.

- **Marks:** Son marcas geográficas ubicadas en el mapa del juego. Cada marca almacena información de latitud y altitud, permitiendo su correcta visualización dentro del juego.
- **UserInfo:** Estructura de información que contiene una lista con los identificadores (ID) de los usuarios activos en el sistema. Esta colección sirve como referencia para operaciones globales sobre los jugadores conectados.
- **GameInfo:** Estructura de información del estado general del juego, incluyendo todos los elementos mencionados anteriormente. Además, contiene parámetros clave como recompensas por combate o número de pasos restantes para realizar combates, entre otros.

Cabe mencionar que algunos de los datos definidos en las clases no llegaron a ser utilizados en la versión final del juego debido a limitaciones de tiempo. Sin embargo, todos estos datos están disponibles en la base de datos y han sido diseñados para ser completamente configurables y editables, lo que permite su integración en futuras ampliaciones del juego. Otro aspecto importante es que las estructuras de UserInfo y GameInfo se generan de manera automática. De esta misma forma, al introducir algún elemento en una colección, su ID se almacena de manera automática en la estructura correspondiente.

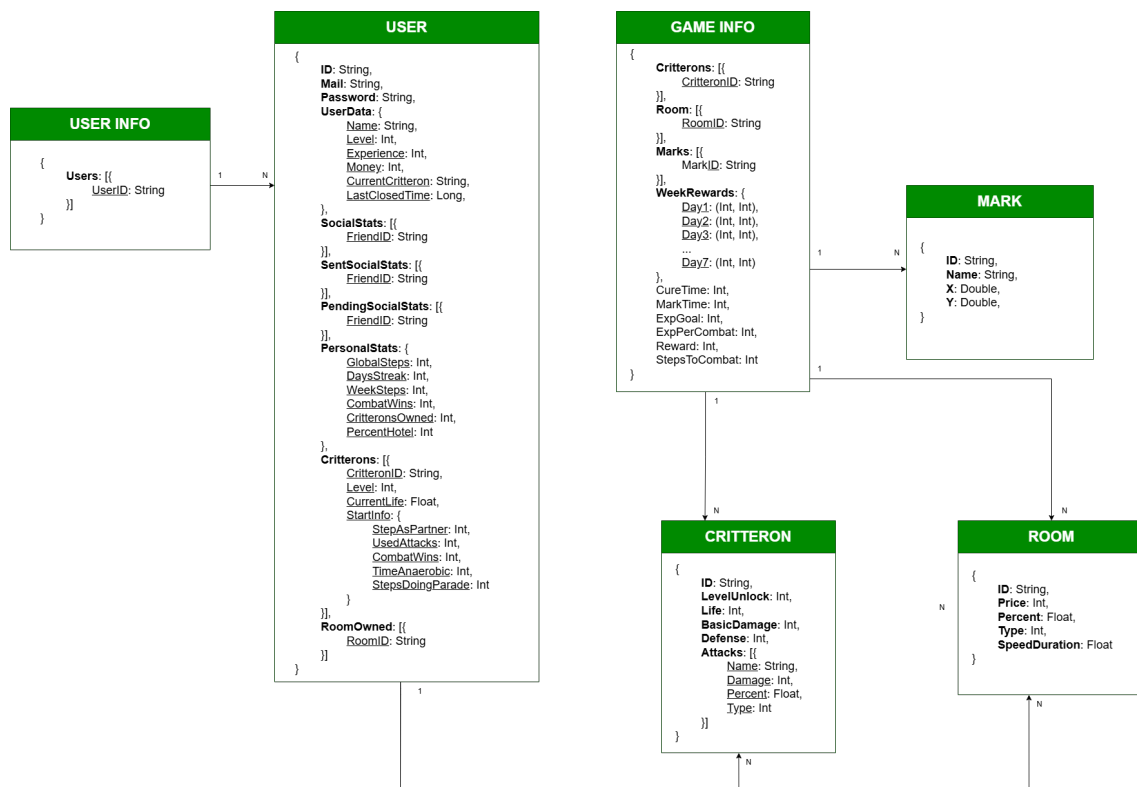


Figura 3.9: Estructura de la base de datos

3.4.2. Almacenamiento de datos: MongoDB

Para el almacenamiento de los datos del juego se ha utilizado *MongoDB*, una base de datos *NoSQL* orientada a documentos.

Las bases de datos *NoSQL* son un tipo de bases de datos que no siguen un modelo basado en tablas. En su lugar, permiten almacenar la información en otros formatos más flexibles como documentos. Este enfoque es especialmente útil para nuestro juego, ya que los datos que almacenamos pueden guardarse en formato *JSON*, el cual representa los datos como objetos formados por pares clave-valor. Esto nos proporciona una gran facilidad a la hora de modificarlos y obtenerlos dentro del juego.

Durante la fase inicial del desarrollo, se realizaron pruebas en local para comprobar su funcionamiento y familiarizarse con su uso. Además, se utilizó *MongoDB Atlas*, la versión en la nube de *MongoDB*. Ambas configuraciones compartían la misma estructura, diferenciándose únicamente en el punto de conexión con la base de datos. Para ello, se creó una pequeña API que controlaba únicamente los *critterons*. Como se explicará en el siguiente apartado más en detalle, esta prueba tenía las siguientes partes:

- **PruebaApplication:** clase inicial que inicia la aplicación. En su método *main* configuramos el contexto de *Spring Boot* y ponemos en marcha el servidor.
- **CritteronsService:** contiene la lógica. Permite realizar operaciones como guardar, buscar y eliminar *critterons* usando el repositorio.
- **CritteronsRepository:** actúa como acceso a la base de datos de *MongoDB*. Esta extiende de *MongoRepository*, lo que nos permite usar sus funciones como hacemos en *CritteronsService*.
- **CritteronsController:** es el encargado de manejar las solicitudes proporcionando endpoints que nos permiten agregar, consultar, actualizar y eliminar *critterons*.
- **Critteron:** clase que define la estructura de documentos que se almacenarán en la colección *Critterons* que hemos creado en *MongoDB*.

3.4.3. API y backend: Spring Boot

El backend del proyecto ha sido desarrollado con *Spring Boot*, más concretamente la versión 3.3.5. Este es un framework de Java que da una gran libertad y flexibilidad a la hora de diseñar y desarrollar aplicaciones. Permite crear aplicaciones web de forma sencilla, conectar bases de datos con proyectos y la creación de backends. Por ello hemos desarrollado en este framework nuestra API REST, la cual es un conjunto de reglas que facilita la comunicación entre diferentes aplicaciones a través

de llamadas HTTP. A continuación, se explica su estructura y funcionamiento de esta.

Para gestionar los datos del juego, se han creado diversas colecciones en MongoDB representando a los diferentes tipos de datos que hemos desarrollado anteriormente. Todos estos datos siguen en nuestro backend una estructura común. Todos ellos tienen una clase propia en la que se definen las variables que van a guardar cada uno de ellos. Además de estos todos tienen una serie de secciones que se adaptan a los datos guardados en las clases anteriormente dichas.

La primera parte es la de los controladores. Estos se encargan de manejar las solicitudes HTTP y definir los puntos de acceso (endpoints) de la API. Es decir, aquí se definen las diferentes llamadas que se hacen para obtener, crear o modificar los datos. Ejemplos de controladores son el `UserController`, que maneja el registro de usuarios, el `CritteronController`, que gestiona las criaturas del juego, y el `GameInfoController`, que maneja los elementos del juego entre otros. Los controladores utilizan los verbos HTTP, los cuales son: GET, que nos permite extraer un documento concreto de la base de datos; POST, que crea uno nuevo; PUT, que actualiza valores de un documento ya existente; y DELETE, que lo borra de la base de datos. Como se ha dicho anteriormente, este flujo de datos da y almacena elementos de tipo JSON. Al realizar una llamada, esta se delega a la siguiente sección, que es la del servicio.

La segunda sección son los servicios. Estas contienen la lógica que ejecuta cada una de las acciones solicitadas por los controladores, es decir, las diferentes llamadas HTTP. Se encargan de coordinar las operaciones de las distintas partes del sistema y de interactuar con la base de datos a través de los repositorios. Por ejemplo, el `UserService` gestiona el registro de usuarios, su progreso, el sistema de amigos y la actualización de los diferentes campos de los usuarios.

La tercera sección son los repositorios, que están implementados como interfaces que extienden `MongoRepository`. Gracias a Spring Boot, la integración con MongoDB es muy sencilla, lo que nos permite que los documentos de la base de datos se conviertan automáticamente en objetos de Java.

La API REST sigue los principios de RESTful, lo que implica una estructura clara de rutas, el uso adecuado del HTTP y respuestas consistentes de las llamadas. Por ejemplo, para gestionar a los *critterons*, hay puntos de acceso como `POST /api/v1/critterons` para crear nuevos de ellos, `GET /api/v1/login` para autenticar a los usuarios, o `GET /api/v1/userTop` para obtener el top 3 de usuarios, entre otros. Entre las funcionalidades que se realizan en la API, aparte del flujo de datos, encontramos las llamadas para obtener los usuarios top debido a la facilidad que tiene para comunicarse con la base de datos de una manera clara y rápida, el sistema de amigos, que permite enviar y aceptar solicitudes, y una vez validada por el otro usuario permite asignarlo como amigo directamente. También se han implementado validaciones, como la verificación de que los correos electrónicos sean de Gmail, y la creación de un control de errores para poder depurar y usar de manera clara la

API.

3.4.4. Seguridad

La seguridad del backend se ha implementado utilizando Spring Security. Este es un módulo de Spring Boot que proporciona autenticación y autorización. Este sistema de seguridad se ha implementado para proteger los datos de los usuarios y garantizar un acceso seguro a los recursos de nuestro juego.

La implementación de la seguridad se basa en varios componentes. El primero es el sistema de autenticación basado en JWT (JSON Web Tokens). Con estos, los usuarios pueden obtener un token único al iniciar sesión. A partir de esto, este token se utiliza posteriormente para verificar la identidad del usuario en cada solicitud que realice durante el juego. El desarrollo de la seguridad se encuentra en la clase `SecurityConfiguration`.

Además, para gestionar el acceso a los recursos, se ha implementado una lista blanca de llamadas que permite el acceso público a ciertos endpoints esenciales para el juego. Entre ellos encontramos la creación de usuarios, el inicio de sesión y algunos endpoints informativos del juego. El resto de los endpoints requieren la autenticación ya mencionada. Para comprobar la veracidad de estos tokens está el `AuthTokenFilter`. Esta clase se encarga de validar los tokens JWT en cada solicitud que se haga. Este filtro examina el token que se encuentra en el header de la petición y verifica su validez y su estado, debido a que estos tokens tienen una caducidad para una mayor seguridad.

Todo este sistema de seguridad trabaja en conjunto con el resto de componentes de la aplicación, asegurando que las operaciones sensibles, como la modificación de datos de usuario o la gestión de elementos del juego, solo puedan ser realizadas por usuarios autorizados.

3.4.5. Automatización y despliegue

Antes de desplegar la API, es fundamental revisar el contenido del archivo `application.properties`, ya que en él se configura, entre otras cosas, la conexión con la base de datos. Esta base de datos puede encontrarse en un entorno local, como ha sido habitual durante el desarrollo de la API y el juego, o en un entorno remoto, accesible a través de la web. Gracias a esta configuración, podemos adaptar fácilmente el entorno según sea necesario para el desarrollo o para el despliegue en producción.

Una vez comprendida la importancia de este archivo, el siguiente paso es ejecutar el archivo `.jar`, que es un paquete que contiene nuestra API, es decir, una aplicación Java lista para ser ejecutada. Este archivo `.jar` ha sido generado mediante Maven, una herramienta de gestión y automatización de proyectos Java que permite compilar el

código, gestionar dependencias y empaquetar la aplicación de una forma eficiente.

Sin embargo, como el proceso de despliegue puede resultar algo lento o repetitivo, se creó un archivo `.bat`, que permite automatizar este procedimiento. Un archivo `.bat` contiene una serie de comandos que se ejecutan de forma secuencial, lo cual facilita, por ejemplo, seleccionar de forma rápida a qué base de datos se va a conectar la API y lanzar su ejecución en un simple paso.

Además, al cambiar de una base de datos a otra, es importante tener en cuenta que existen ciertos archivos que son esenciales para el funcionamiento del juego. Elementos como los *critterons*, las habitaciones o las marcas forman parte del contenido inmutable que debe estar disponible desde el inicio. Por ello, se creó un segundo archivo `.jar`, correspondiente a una aplicación encargada de cargar estos datos esenciales en la base de datos a través de nuestra API. De esta manera, nos aseguramos de que la base de datos cuente con toda la información necesaria para que el juego funcione correctamente desde el primer momento.

3.5. Diseño y estudio del plan de pruebas

3.5.1. Diseño del estudio y selección de participantes

Teniendo en cuenta los estudios previos, hemos creado un plan de pruebas adaptado a nuestro juego, buscando comprobar el impacto del juego en los hábitos de actividad física de los usuarios y, al mismo tiempo, detectar posibles fallos o mejoras en el diseño.

El grupo de participantes estará formado por unas 20 personas, intentando que haya la mayor variedad posible en cuanto a edad, género y hábitos de ejercicio, para observar cómo reacciona cada perfil al uso del juego. Además, por cada persona que utilice la aplicación habrá otra que irá rellenando un documento de seguimiento de su actividad física durante el tiempo de pruebas, sin utilizar la app. Esta comparación directa permitirá entender mejor el posible impacto del juego entre las personas que lo usan y las que no. La prueba durará como mínimo una semana, y se irá aumentando el tiempo de prueba a medida que se vaya observando el impacto de este en los jugadores.

3.5.2. Objetivos del plan de pruebas

Los principales objetivos de esta prueba son, por un lado, valorar si el uso de la aplicación tiene una repercusión positiva en la cantidad y duración de las sesiones de ejercicio, y por otro, observar si puede influir en los hábitos de los participantes. Para ello, se quiere responder a las siguientes preguntas:

- ¿Durante el uso del juego, existe una repercusión positiva en el número y tiempo de las sesiones de ejercicio?
- ¿Existen diferencias entre los diferentes perfiles de usuario?
- ¿Es el juego capaz de influir en los hábitos de ejercicio de las personas?

3.5.3. Metodología de la prueba

La prueba seguirá una serie de pasos. Antes de comenzar, cada participante completará un cuestionario inicial para conocer su situación actual respecto al ejercicio físico. En este se realizarán preguntas del mes previo a la prueba. De esta manera podemos conocer qué tipo de ejercicio hace, durante cuánto tiempo, y cómo lo realiza. Además se le preguntará si utiliza dispositivos que le ayuden en la realización de este y si ha jugado anteriormente a un exergame. Posteriormente se les facilitarán las credenciales para acceder al juego a los jugadores que van a probar el juego y se explicará el ciclo de este y sus mecánicas. A las personas que realicen el cuestionario diario se les informará de la hora del día que llegará el correo electrónico con el cuestionario a realizar. Este será un cuestionario corto sobre si han realizado ejercicio durante ese día, y en caso afirmativo se hará una serie de preguntas para saber de qué tipo. Durante la semana de prueba, se estará disponible para resolver posibles dudas de los participantes. Mientras tanto, se hará un seguimiento de los datos recogidos por la base de datos, descargando los registros correspondientes a cada usuario para evaluar su evolución. Al finalizar se realizará una entrevista breve para recoger impresiones generales. Además, una semana después, se les enviará un cuestionario final para comprobar si el juego ha tenido algún efecto duradero.

Las entrevistas se realizaron de forma individual y de manera remota. A continuación, se presentan las preguntas utilizadas como base para la entrevista, aunque cabe destacar que estas preguntas sirvieron como guía para mantener una estructura durante la entrevista ya tuvimos la libertad de adaptar o formular nuevas preguntas en función de las respuestas y del perfil de cada participante:

- ¿Has aumentado el tiempo y número de sesiones de ejercicios durante esta última semana? Si es así, ¿en qué medida?
- ¿Crees que el juego ha tenido un aspecto positivo en esto? ¿Por qué?
- ¿Cómo te has sentido mientras usabas el juego? ¿Te ha resultado fácil de entender y usar?
- ¿Cómo describirías tu experiencia general con el juego? ¿Hubo algo que te gustó especialmente o que no te gustó? ¿El qué?
- ¿Sentiste que tu progreso en el juego fue satisfactorio? ¿Hubo momentos en los que sentiste que tu progreso se estancaba? ¿Cuáles?

- ¿Te resultó interesante desbloquear nuevos *critterons* a medida que subías de nivel?
- ¿Te gustaron las interacciones sociales del juego? ¿Competir o colaborar con otros jugadores te pareció divertido?
- ¿Te motivó el sistema de rankings semanales a seguir jugando más tiempo?
- ¿Hubo algún momento en el que perdiste interés en el juego? ¿Qué crees que lo causó?

3.5.4. Métricas e interpretación de resultados

Durante la prueba, se han tenido en cuenta métricas como los pasos diarios, número de *critterons* obtenidos, nivel alcanzado, amigos añadidos o habitaciones compradas. La entrevista posterior ha ayudado a interpretar estos datos y entender la experiencia del usuario, preguntando por aspectos como la facilidad de uso, la motivación, la progresión o las interacciones sociales dentro del juego. También durante la entrevista se han realizado preguntas para detectar si se ha sentido motivado, si ha encontrado dificultades técnicas o de diseño y si ha interactuado con otros usuarios. Esto nos ha permitido ver de forma más clara qué efecto ha tenido el juego en cada perfil.

3.5.5. Implementación de los cuestionarios y herramientas de seguimiento

Los cuestionarios que se usan en las pruebas se implementaron en Google Forms, dada su facilidad de uso y la familiaridad de los usuarios con esta herramienta. La distribución de los cuestionarios diarios se realizó de forma automática mediante Google Docs, utilizando una hoja de cálculo con los correos electrónicos de los participantes y la extensión de Apps Script para el envío programado. De este modo, se garantiza una forma ágil y constante de recolección de datos a lo largo de la semana de pruebas. A su vez, cada día durante las pruebas se hacía una descarga de los datos relativos a los usuarios desde la base de datos de MongoDB para, posteriormente, realizar un análisis más detallado de los datos de los usuarios y cómo varían estos de un día a otro. Se realizó todos los días a la misma hora para mantener una coherencia entre los datos.

3.6. Desarrollo y resultados de las pruebas

3.6.1. Grupo de participantes

El grupo de participantes ha estado compuesto por un total de 21 personas. Para realizar una comparación directa, los participantes se dividieron en dos grupos: 9 personas utilizaron el juego, mientras que 12 realizaron un seguimiento de su actividad física diaria mediante cuestionarios, sin utilizar la aplicación. Esta disparidad fue debida a la incompatibilidad de algunos usuarios tanto a nivel técnico como de tiempo. La diversidad de los usuarios ha sido variada en cuanto a edad, sexo y hábitos de ejercicio. Para ello los estudiaremos más a detalle a partir del cuestionario que realizaron antes de realizar las pruebas.

En cuanto a la distribución por sexo de los usuarios, se observa un reparto casi equitativo entre hombres y mujeres (Figura 3.10). En la parte de la edad, destacan dos grupos principales: los usuarios más jóvenes, con edades entre los 15 y los 34 años. Estos son un total de 13 personas, y los mayores de 55 años, con 7 participantes en los rangos de 55 a 64 años, lo que refleja una menor representación de edades intermedias (Figura 3.11). En relación con la frecuencia de ejercicio, más de la mitad de los usuarios realiza actividad física de forma ocasional, siendo lo máximo que se realiza entre una y dos veces por semana, mientras que los 8 usuarios restantes muestran una rutina mucho más constante, incluyendo 2 personas que realizan ejercicio a diario (Figura 3.12). En cuanto al tiempo, casi la mitad de los usuarios le dedican entre media y una hora. Y casi ninguno de ellos le dedica menos de 15 minutos (Figura 3.13).

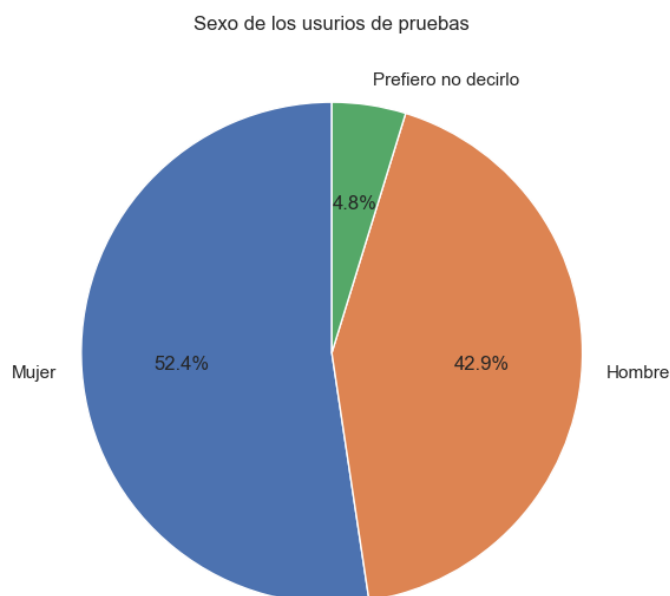


Figura 3.10: Sexo de los usuarios

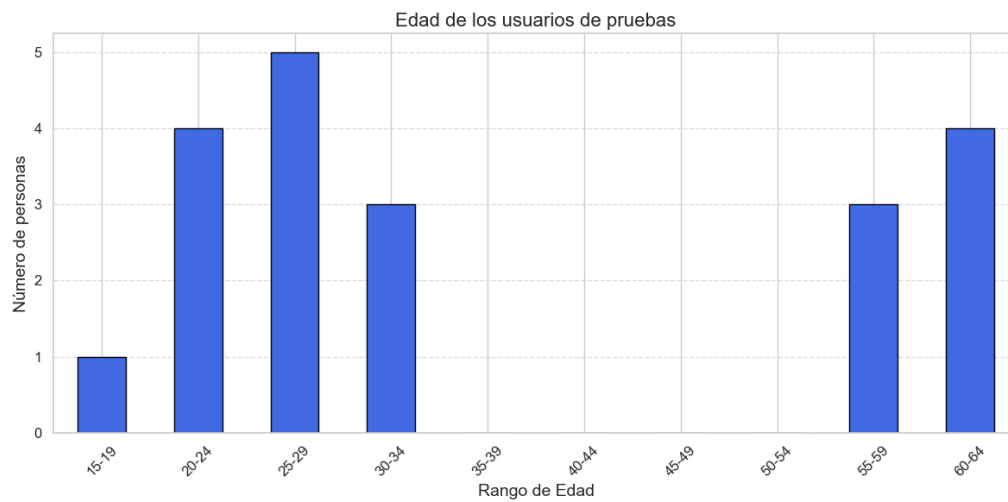


Figura 3.11: Edad de los usuarios

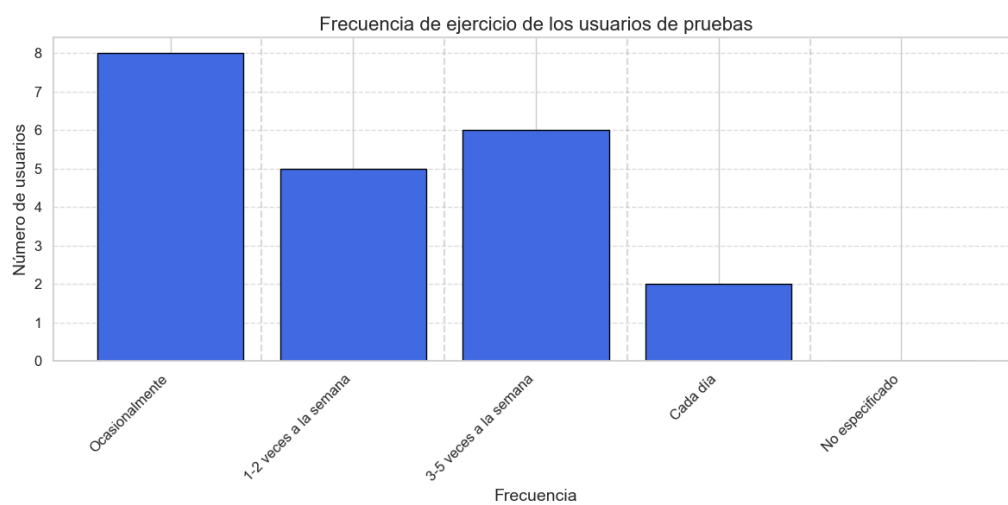


Figura 3.12: Frecuencia de ejercicio de los usuarios

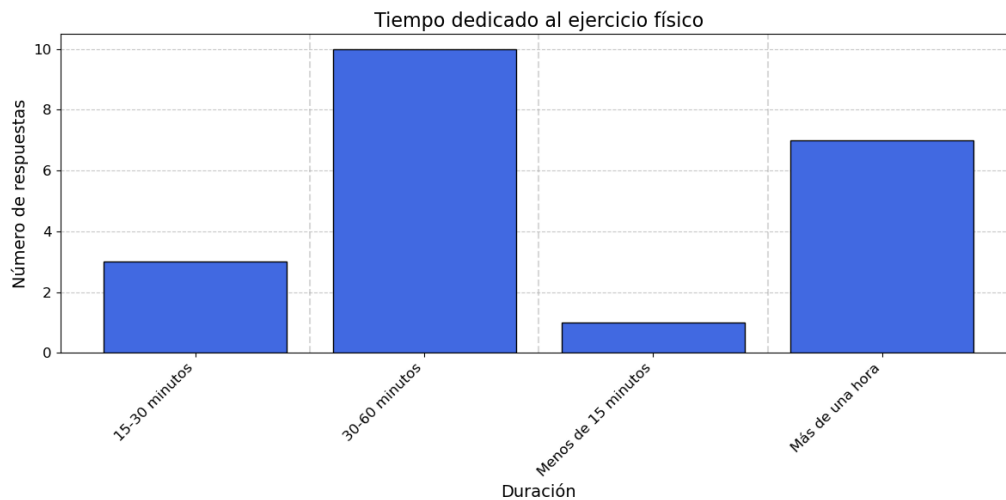


Figura 3.13: Tiempo de ejercicio de los usuarios

De manera general, se observa una tendencia por las actividades físicas de baja intensidad, como caminar o dar paseos, la cual es opción más frecuente entre los usuarios de pruebas. Le siguen ejercicios de cardio, especialmente aquellos realizados al aire libre o en grupo, como correr y bailar (Figura 3.15). Por último, aunque con menor participación, también se practican entrenamientos de fuerza o deportes en equipo (Figura 3.14). Es por ello que el grupo de usuarios se adapta de una manera correcta a las pruebas ya que la gran mayoría de ellos incorporan los paseos o correr en sus sesiones de ejercicios. Sabiendo esto los usuarios también fueron divididos respecto a este criterio, priorizando a aquellos que lo incorporaban en su rutina de una manera más común.



Figura 3.14: Tipo de ejercicio realizado por los usuarios



Figura 3.15: Ejercicios de cardio realizado por los usuarios

En cuanto a la manera de realizar ejercicio por parte de los usuarios encontramos una gran variedad. Mientras que algunos lo hacen en solitario, otros prefieren compartir la actividad con amigos o familiares. También hay quienes adaptan su elección en función del día, lo que sugiere una rutina variable, alternando el tipo de ejercicio que realizan. La menor parte lo realiza en grupo. Debido a la gran diversidad, entre los usuarios de prueba del juego estarán los que prefieren realizar la actividad con amigos o familiares. De esta manera habrá usuarios conscientes de que otros usuarios conocidos para ellos también están jugando al juego, incentivando la socialización dentro de este (Figura 3.16).

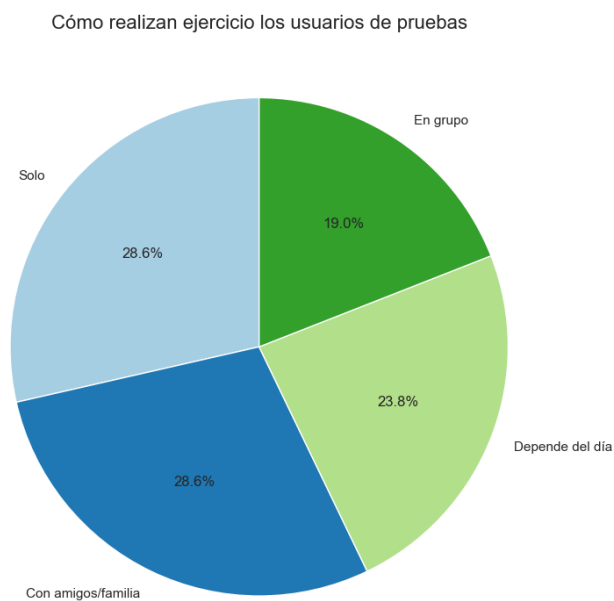


Figura 3.16: Cómo realizan el ejercicio los usuarios

En cuanto a los espacios donde los usuarios realizan ejercicio, el gimnasio aparece como el entorno más habitual, seguido por zonas urbanas. También se utilizan entornos más naturales, como senderos o parques, aunque en menor medida. Esto se debe a la ubicación de los usuarios. Geográficamente gran parte se encuentra viviendo en ciudades como Madrid, Sevilla y Las Palmas. Mientras que otra son de un pueblo de Córdoba. Por ello esto es un factor clave en saber donde se realizan los ejercicios, sobre todo los realizados al aire libre. Además otros lo hacen en salones de baile o pistas deportivas, siendo esto específico a los tipos de ejercicio de cada usuario (Figura 3.17).

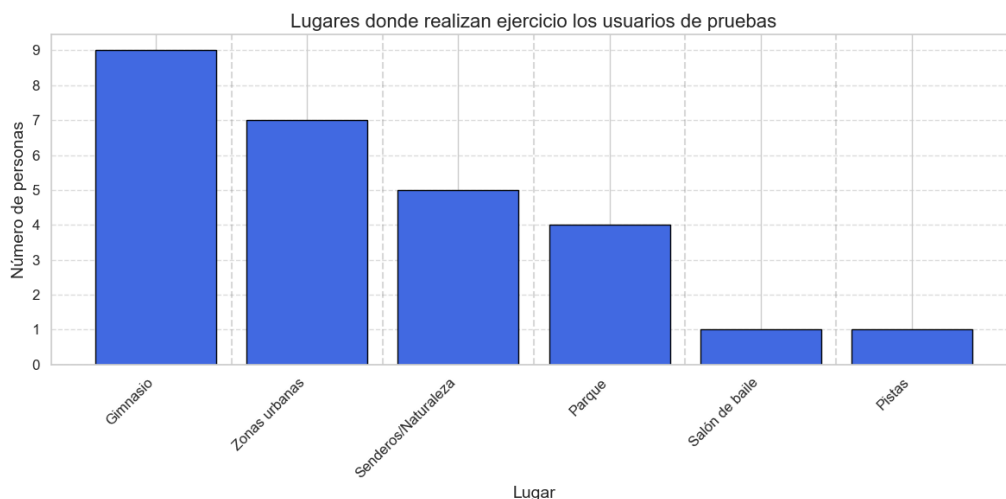


Figura 3.17: Dónde realizan el ejercicio los usuarios

En lo que respecta a la motivación para hacer ejercicio, la mayoría se sitúa en niveles medios-altos. Aunque también existen los casos en los cuales realizar ejercicio no es una actividad que motive a algunos usuarios (Figura 3.18). Las razones que impulsan esta motivación son principalmente personales, destacando el mantenerse en forma como la razón principal para ello. Otras motivaciones como socializar, reducir el estrés también están presentes, aunque con menor frecuencia. Esto sugiere que la realización de ejercicio está fuertemente influenciado por metas individuales o necesidades médicas (Figura 3.19).



Figura 3.18: Motivación por hacer ejercicio de usuarios

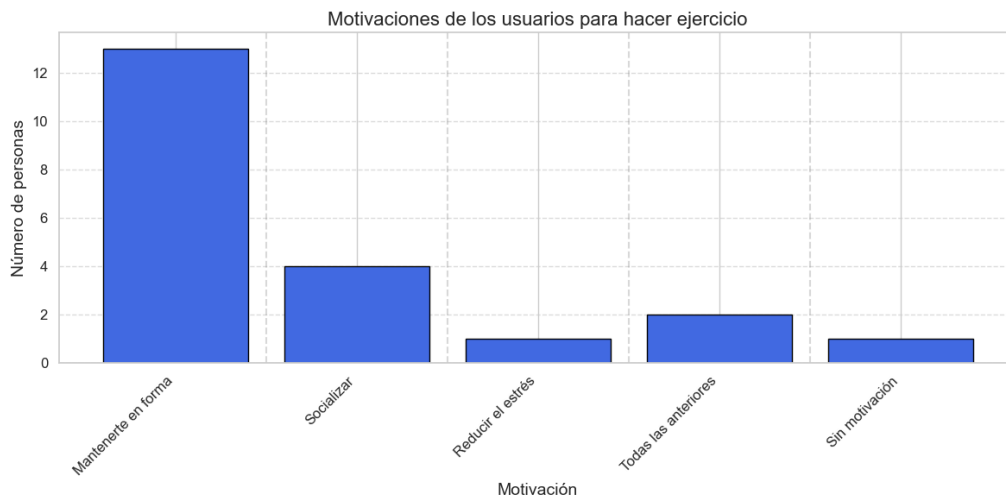


Figura 3.19: Motivo para realizar ejercicio de los usuarios

Por otra parte, aunque el grupo de usuarios tiene hábitos de ejercicio más o menos asentados, el uso de aplicaciones o dispositivos durante sus sesiones de ejercicios aún no juega un papel relevante. La mayoría no los utiliza para monitorizar o acompañar sus sesiones, lo que puede estar relacionado tanto con preferencias personales como con barreras de acceso o desconocimiento. Solo un pequeño porcentaje los usa y estos son pulseras o relojes inteligentes o el uso de aplicaciones de seguimiento de la actividad física (Figura 3.20).

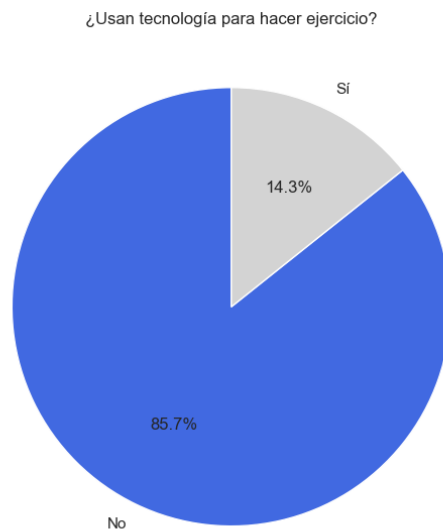


Figura 3.20: Usuarios que usan tecnologías durante los ejercicios

Por último, a diferencia del uso de aplicaciones o dispositivos durante las sesiones de ejercicio, una parte significativa de los usuarios ha jugado a exergames, aunque la mayoría no lo ha hecho nunca. Esto implica que aún no se ha consolidado como una opción común dentro de sus entornos (Figura 3.21). Entre los juegos más jugados destacan *Just Dance*, *Wii Sport* o *Pokemon Go!*.

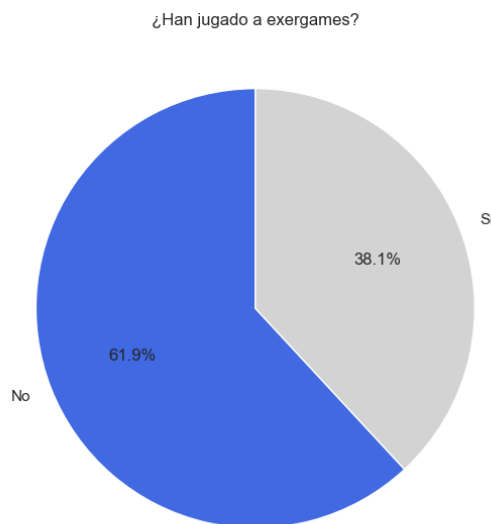


Figura 3.21: Usuarios que han jugado a exergames

Como se ha podido comprobar, el grupo de usuarios de prueba es diverso, lo cual permite la validez y fiabilidad de los datos recogidos mediante el plan de pruebas, gracias a que se incorporan diferentes perfiles dentro de este.

3.6.2. Desarrollo de la prueba semanal

Una vez completado el cuestionario inicial, se facilitó a los usuarios que prueban el juego las credenciales de acceso a este, junto con una breve explicación sobre su funcionamiento. A los participantes que realizaron el seguimiento mediante formularios se les informó del horario en el que recibirán diariamente el enlace al cuestionario. Durante la semana se estuvo disponible para resolver las posibles dudas de los participantes y asegurar el correcto desarrollo del estudio. A lo largo del proceso se publicaron dos actualizaciones con correcciones menores, debido a errores detectados durante las pruebas. Estas actualizaciones abordaron principalmente problemas relacionados con la sincronización con la base de datos. Dado que la subida de datos depende de factores externos, como la conexión a internet o la latencia del servidor, se incorporaron pantallas de carga para evitar desajustes provocados por estos retrasos. Además, se mejoró la visualización de la interfaz de usuario y se corrigió un problema relacionado con las animaciones y el movimiento de los *critterons* dentro del hotel.

3.6.3. Resultados de las métricas del juego

Las pruebas se llevaron a cabo durante ocho días seguidos. Hay que tener en cuenta ciertos aspectos que influyen directamente en los datos. Primero, algunos usuarios, como se verá más adelante en las entrevistas, no podían seguir realizando las pruebas el octavo día, influyendo esto de una manera directa en los datos como se observará. Por otra parte, la extracción de datos de los usuarios de la base de usuarios se realizó cada día a las 22:00. Por último, las pruebas coincidieron con una semana festiva, lo que pudo influir en los patrones de actividad de los usuarios.

Antes de ver cómo son las gráficas que se han obtenido del juego, primero tenemos que ver qué usuarios han sido los que van a realizar estas pruebas. Este grupo fue elegido dependiendo de varios criterios. El primero es la viabilidad de los usuarios. Algunos de los usuarios que se inscribieron nos informaron que solo podrían participar en la realización del cuestionario diario, por lo que no formaron parte de la selección. Primero se organizaron por grupos según los hábitos de ejercicio. Se eligieron primero usuarios que hacían ejercicio de una manera más rutinaria y a su contraparte, que lo realizaban ocasionalmente o 1 o 2 veces a la semana. Después debido a que un gran grupo tiene hábitos muy parecidos se priorizaron los que entre sus motivos para hacer ejercicio estaba socializar. Tras esto se eligieron los que tenían los paseos y las actividades de cardio como tipo de ejercicio que realizaban, mezclando a los que incorporan otro tipo de ejercicios. Un factor importante tener usuarios de diferentes rangos de edad así como tener usuarios de distinto sexo de una manera balanceada (Tabla 3.4).

Edad	Género	Frecuencia	Ejercicio	Objetivo	Color
22	Mujer	1-2 días semanales	Cardio, Entrenamiento de fuerza	Mantenerse en forma	
23	Mujer	1-2 días semanales	Cardio, Deportes en equipo	Mantenerse en forma	
25	Hombre	3-5 días semanales	Cardio, Flexibilidad y equilibrio	Mantenerse en forma	
26	Hombre	Ocasionalmente	Deportes en equipo	Mantenerse en forma, Socializar	
26	Mujer	Ocasionalmente	Andar / Paseos	Mantenerse en forma	
30	Hombre	Ocasionalmente	Andar / Paseos	Mantenerse en forma, Socializar	
33	Hombre	Ocasionalmente	Deportes en equipo	Mantenerse en forma, Socializar	
55	Mujer	3-5 días semanales	Andar/Paseos, Flexibilidad y equilibrio	Mantenerse en forma, Socializar	
58	Mujer	1-2 días semanales	Cardio, Flexibilidad y equilibrio	Mantenerse en forma	

Tabla 3.4: Características de los usuarios

Una vez conocido el tipo de usuarios que han realizado las pruebas, pasaremos a ver cuales han sido los resultados de las variables del juego. Tenemos que tener en cuenta que tantos los días 4 y 5 fueron festivos nacionales, por lo que esto puede influir en el estudio. Primero veremos el uso de la aplicación mientras se andaba. Como se puede observar, durante los primeros tres días la mayoría de los usuarios no superó los 10.000 pasos diarios. A partir de entonces, un grupo de ellos comenzó a aumentar progresivamente su actividad, llegando incluso a los 20.000 pasos en los días posteriores. Sin embargo, la mayoría de los participantes nunca alcanzaron dicha cifra incluso al terminar las pruebas (Figura 3.22). Además observando el incremento de pasos globales podemos ver como la mayoría de ellos mantuvo un número constante sobre todo durante los primeros días. Sin embargo, hay usuarios que no entran en este patrón. Por ello podemos distinguir dos perfiles: quienes man-

tuvieron un ritmo de pasos constante y quienes, día a día, incrementan su actividad, en ocasiones sumando hasta 15.000 pasos extra (Figura 3.23). Cabe destacar que un gran número de usuarios abandonó las pruebas en el octavo día. Como se comentó anteriormente, esto no está relacionado con el juego sino con incompatibilidad de tiempo de los usuarios en la mayoría de los casos. Por ello podemos ver que aunque el número de pasos diarios no es elevado sí se ha mantenido constante durante las pruebas.

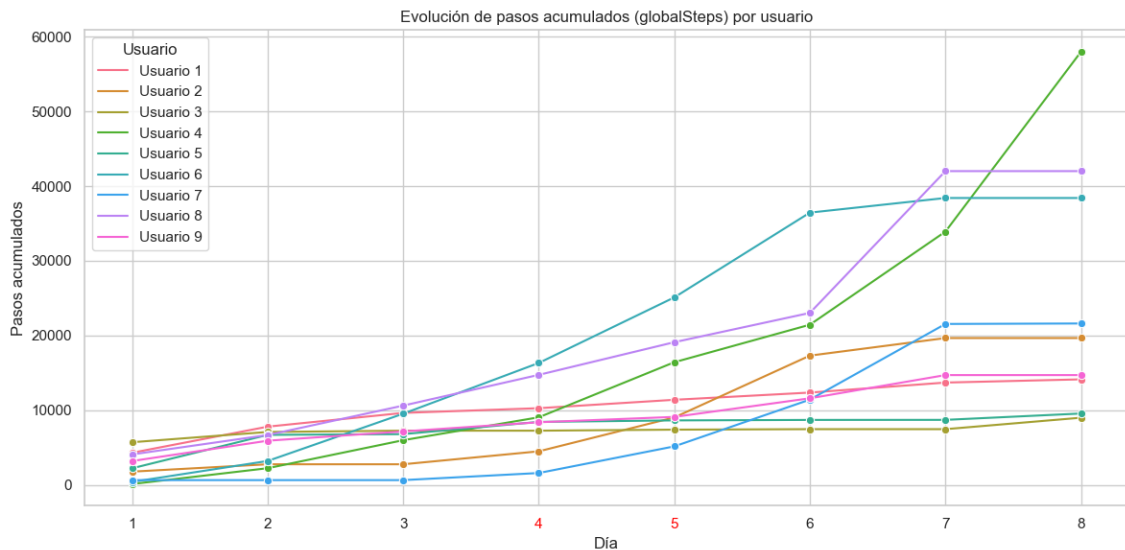


Figura 3.22: Pasos globales de los usuarios

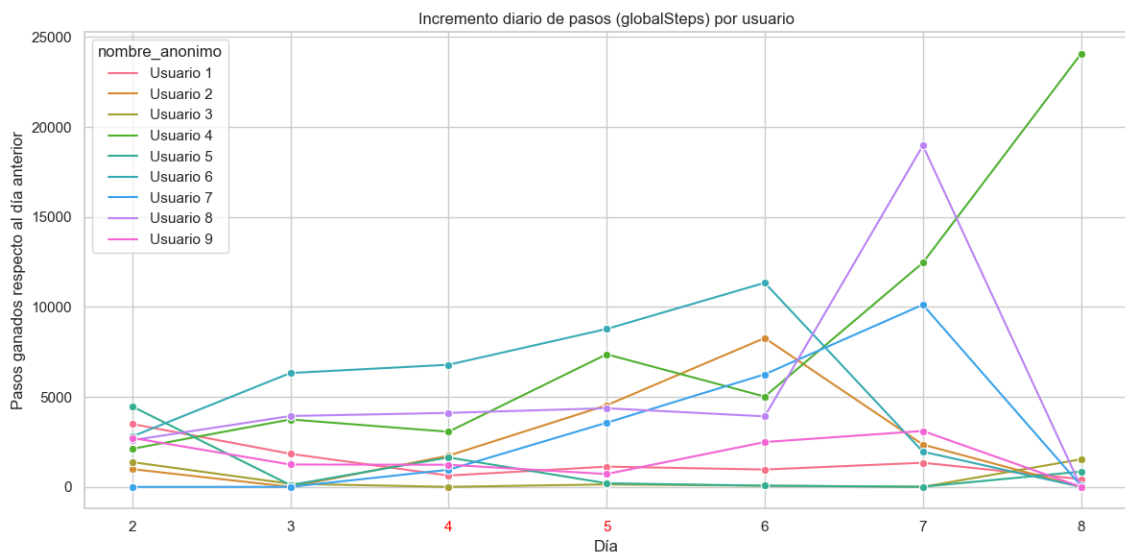


Figura 3.23: Incremento de los pasos por día de los usuarios

En cuanto al uso del juego podemos ver los datos recogidos sobre los niveles alcanzados, el número de *critterons* obtenidos y el número de habitaciones compradas

por los usuarios durante las pruebas. Primero en cuanto al nivel podemos observar que durante los primeros días casi todos los usuarios mantuvieron una constancia, subiendo de 2 a 3 niveles por día. Sin embargo, al igual que en el número de pasos, varios usuarios aumentaron el nivel respecto al día anterior de una manera mucho más marcada que el resto (Figura 3.25). De esta manera el nivel medio conseguido por los usuarios fue el nivel 16. El nivel más alto alcanzado fue el 26 y el usuario que alcanzó menos nivel se quedó en el nivel 5 (Figura 3.24). Siguiendo con los *critterons* conseguidos cabe destacar que solo un usuario consiguió los 12 *critterons* disponibles. Se observa como la mayoría de los usuarios obtuvieron los *critterons*, entre 1 y 2, de manera progresiva a lo largo de los días (Figura 3.26). De esta manera se puede establecer una clara relación entre el nivel obtenido por el usuario y el número de *critterons* obtenidos. En cuanto a las habitaciones observamos cómo cada usuario avanzó de una manera propia, aunque existe la tendencia de una habitación por día durante los primeros días de pruebas. Sin embargo existen casos extremos como el de un usuario que consiguió todas las habitaciones durante los 2 primeros días. Se puede ver como la mitad de los usuarios consiguieron las 6 habitaciones antes del octavo día (Figura 3.27).

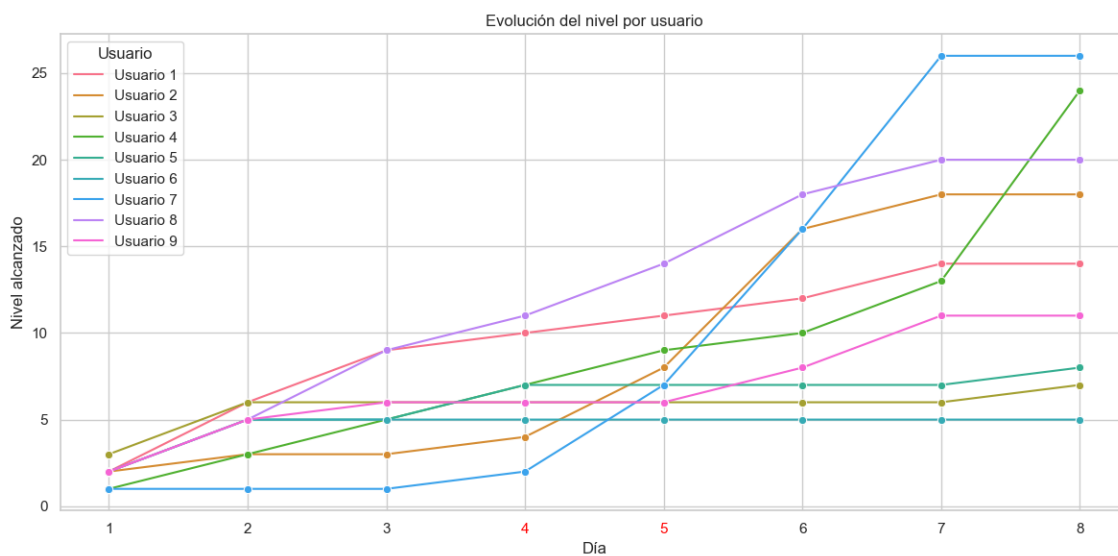


Figura 3.24: Niveles de los usuarios

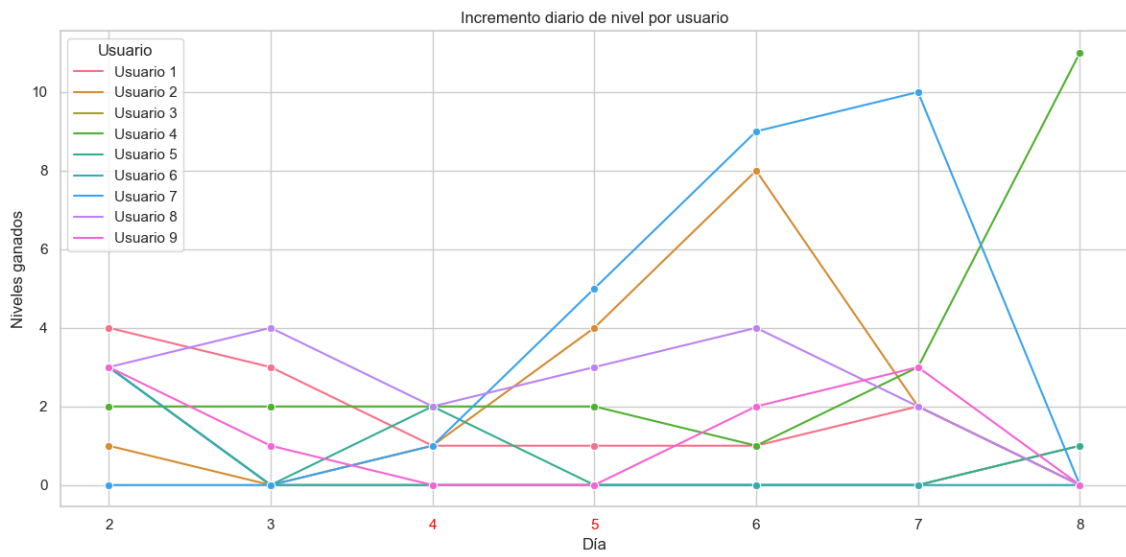


Figura 3.25: Incremento de los niveles por día de los usuarios

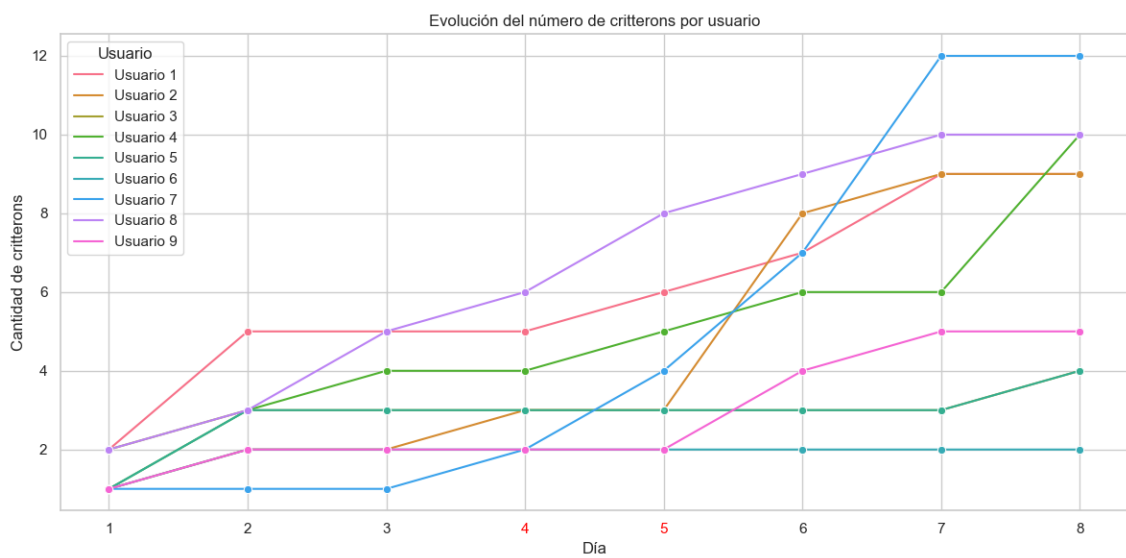


Figura 3.26: Número de critterons de los usuarios

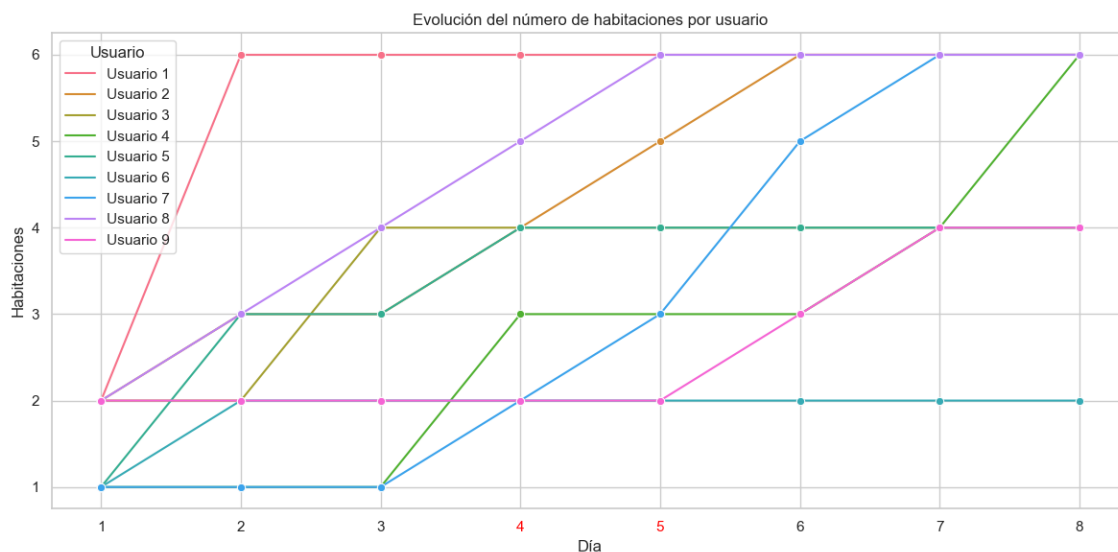


Figura 3.27: Números de habitaciones de los usuarios

Cabe recalcar varios casos interesantes antes de dar una primera visión de cómo se han desarrollado las pruebas y las conclusiones obtenidas. Podemos observar que no todos los usuarios con más pasos son los que más han interactuado con la aplicación. Uno de los que ha conseguido más pasos no ha subido de nivel ni compró habitaciones a partir del segundo día. Por lo que este usuario solo interactuó con la aplicación para contar sus pasos. De esta manera como se ha visto muchos usuarios empezaron a tener una mayor interacción con el juego a partir del tercer/cuarto día.

Para comprender mejor cómo han interactuado los usuarios con la aplicación analizaremos ahora de una manera general a partir de los archivos `traces.log` generados por cada usuario. Primero empezemos por cómo se han realizado los combates en el juego. En total se han realizado 520, siendo más de la mitad de estos combates individuales. El siguiente tipo de combate fue el combate junto a un amigo, y las amistosas las menos, siendo estas únicamente 14 (Figura 3.28). También se puede ver que de estos combates solo se ganaron 308, poco más de la mitad (Figura 3.29).

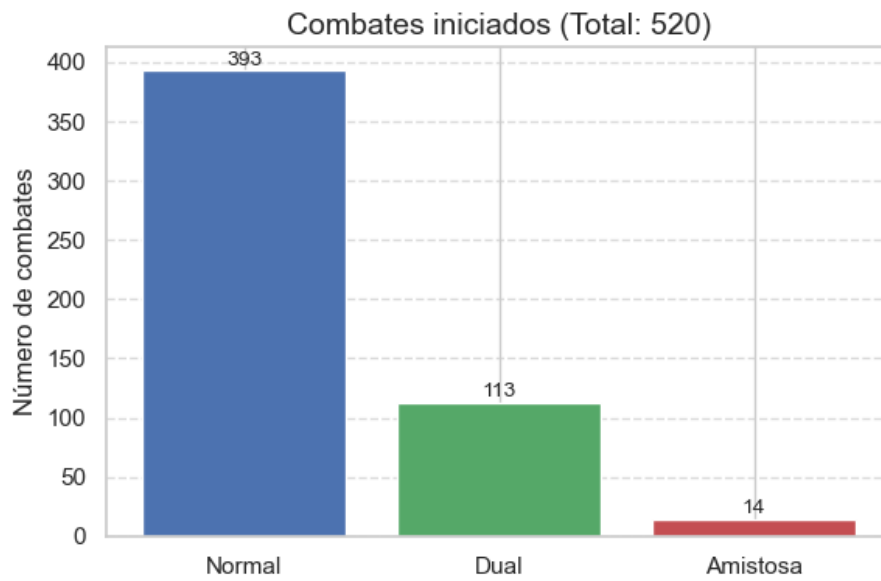


Figura 3.28: Tipos de combates realizados por los usuarios

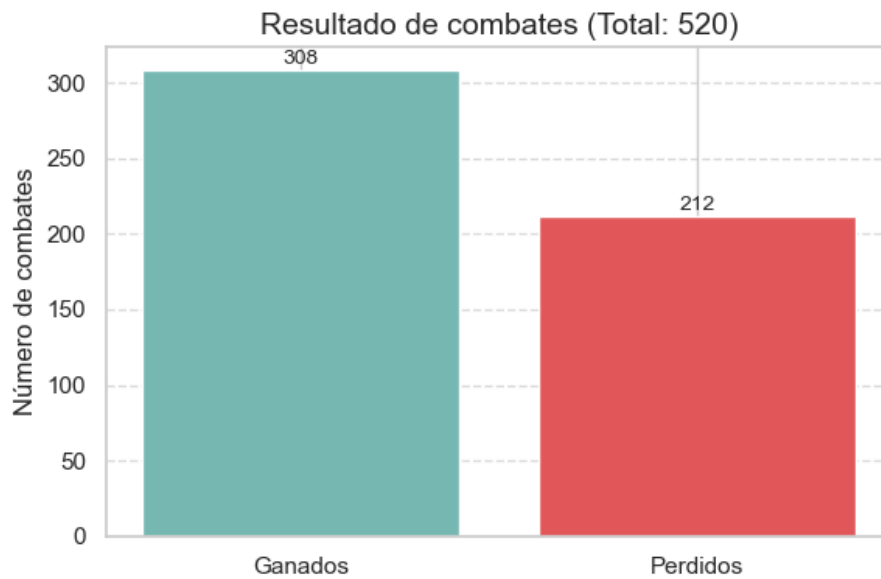


Figura 3.29: Victorias y derrotas de los usuarios

Otro punto importante a destacar son los amigos dentro del juego. Podemos ver que únicamente un usuario no obtuvo amigos y el máximo de amigos por usuario fue de 2 (Figura 3.30). Esto se ve claramente afectado por las conexiones previas de los usuarios y el lugar donde se encontraban durante las pruebas debido a ser un grupo de usuarios reducido.

Red de amistades

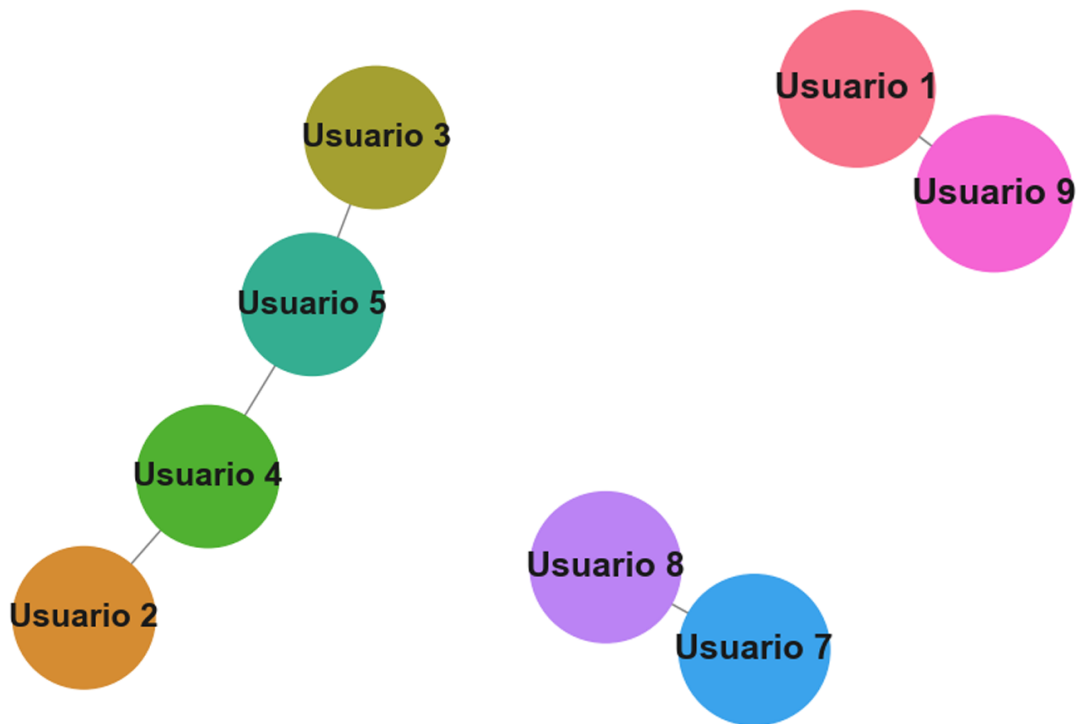


Figura 3.30: Uniones de amistad entre los usuarios

Aunque este apartado nos proporciona información sobre el correcto funcionamiento del juego, también es importante conocer cuántas veces se ha obtenido cada *critteron* por parte de los usuarios. Debido al componente de azar en su obtención, los *critterons* que requieren un nivel más bajo para ser desbloqueados son los más obtenidos. En la gráfica se puede observar cómo estos *critterons* han sido obtenidos en repetidas ocasiones, lo que contribuye a su subida de nivel. En particular, el *critteron* más obtenido ha sido *Spike* (Figura 3.31). Esto se debe a que es el *critteron* inicial con el que comienzan todos los usuarios, y está desbloqueado desde el inicio del juego.

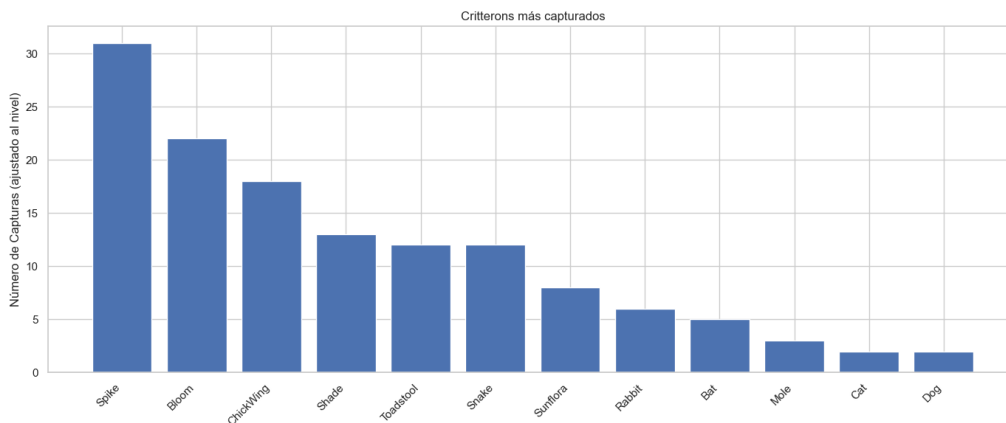


Figura 3.31: Número de veces que se ha obtenido cada critteron

Tras esto podemos diferenciar varios tipos de usuarios. Primero veamos los grupos según interactúan con el juego:

- **Usuario comprometido y activo.** La principal característica es que realiza actividad física y usa el juego de una manera constante. Sube niveles con rapidez, desbloquea habitaciones y captura *critterons*.
- **Usuario físicamente activo pero poco implicado con el juego.** Este tipo de usuario camina mucho y hace ejercicio con frecuencia, pero apenas interactúa con el juego.
- **Usuario desenganchado o inactivo.** Este tipo de usuario abandona el juego poco después de comenzar, sin apenas progresar ni en pasos ni en el juego.

También se puede ver otros tipos de usuarios en cómo obtienen pasos en el juego. Para ver mejor esto se mostrarán unos pequeños mapas de las rutas realizadas por algunos de los usuarios. Los puntos se muestran cada 3 minutos mientras se mantiene la aplicación abierta, por lo que la distancia entre puntos oscila entre los 3 minutos y los 7 minutos de manera general.

- **Usuario que realiza rutas y recorridos de forma lúdica.** Este perfil de usuario utiliza el juego mientras camina con el objetivo principal de realizar ejercicio o seguir una ruta planificada. Es decir, el juego sirve como acompañamiento a su actividad física. En la Figura 3.32 se muestra un ejemplo donde se observa claramente un recorrido circular: el usuario parte de un punto, se desplaza por un trayecto definido y regresa al mismo lugar.

3.6.4. Resultados de la entrevista y el cuestionario final

Finalizada la semana, se realizaron entrevistas individuales para recoger impresiones generales sobre la experiencia, el uso y las interacciones dentro del juego. Además, se preguntó a los usuarios por aspectos como la facilidad de uso o la motivación, entre otros. Una semana después, se envió un cuestionario final para evaluar si el uso del juego había tenido algún efecto duradero en los hábitos de ejercicio de los participantes.

Primero en cuanto a la entrevista en general, la mayoría de los usuarios aumentaron su ejercicio gracias al juego, añadiendo los paseos a su rutina diaria o incrementando el tiempo que le dedicaban. El juego tuvo un impacto positivo principalmente por su sistema de recompensas, donde desbloquear nuevos *critterons* y habitaciones motivaba a seguir caminando. Aunque algunos usuarios tuvieron dificultades al principio para entender las mecánicas del combate o el funcionamiento general del juego, en la mayoría de los casos, una vez familiarizados con este, la experiencia fue valorada como positiva. El aspecto estético del juego, especialmente el diseño de los *critterons* y la posibilidad de coleccionarlos, fue uno de los elementos que más gustó. Sin embargo, cuando los usuarios completaron casi todos los objetivos disponibles, como conseguir la mayoría de *critterons* o comprar todas las habitaciones, algunos perdieron parte del interés, sintiendo que ya no quedaban recompensas que alcanzar que les motivara a seguir caminando. Las interacciones sociales y el sistema de ranking fueron vistos como puntos claves para quienes eran más competitivos, fomentando la implicación y el esfuerzo por mejorar en el juego. Por otro lado, quienes no se consideran competitivos o no entendieron del todo el flujo del juego, no encontraron en estos elementos una fuente de motivación. Aunque hubo algunos usuarios que no sintieron que el juego incentiva lo suficiente el ejercicio físico o la progresión debido a la falta de comprensión de algunos sistemas clave del juego, la percepción general ha sido positiva, y varios usuarios comentaron el interés en seguir utilizando este tipo de juegos en el futuro como complemento para el ejercicio físico o simplemente para el entretenimiento.

Tras la semana de uso del juego, se realizó un último cuestionario para evaluar cómo han cambiado los hábitos de ejercicio de los participantes, en comparación con los datos recogidos antes de las pruebas. Uno de los aspectos más destacados ha sido el aumento en la frecuencia de las sesiones de ejercicio. En el cuestionario inicial, la mayoría de los usuarios indicaba que realizaban ejercicio solo de manera ocasional, sin seguir una rutina. Sin embargo, los resultados actuales muestran un cambio significativo, ya que una gran parte de los usuarios ha comenzado a realizar actividad física con mayor regularidad. En concreto, se observa que la mayoría de los participantes ahora realiza ejercicio entre 3 y 5 veces por semana (Figura 3.34).

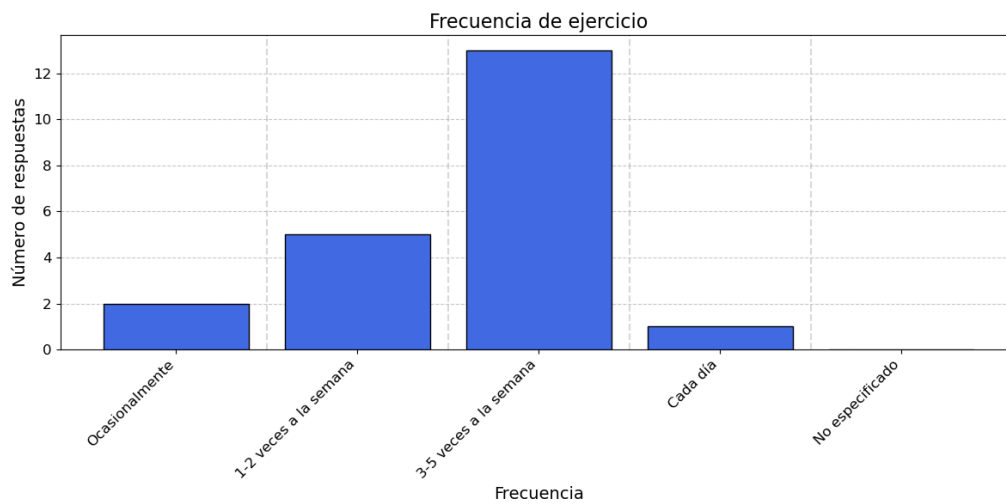


Figura 3.34: Frecuencia de ejercicio de los usuarios tras la semana de pruebas

En lo que respecta al tipo de ejercicio realizado por los usuarios, los resultados indican que andar y los paseos continúan siendo la forma más habitual de actividad física. Esto ya se observaba en el cuestionario inicial, por lo que se mantiene de forma constante dentro de los hábitos de ejercicio de los usuarios. No obstante, cabe señalar que, aunque el tipo de ejercicio se mantiene, ha habido un aumento en el número total de usuarios que lo practican. Esto indica que, además de mantener sus actividades, más personas se han sumado a la práctica de este tipo de ejercicio (Figura 3.35).

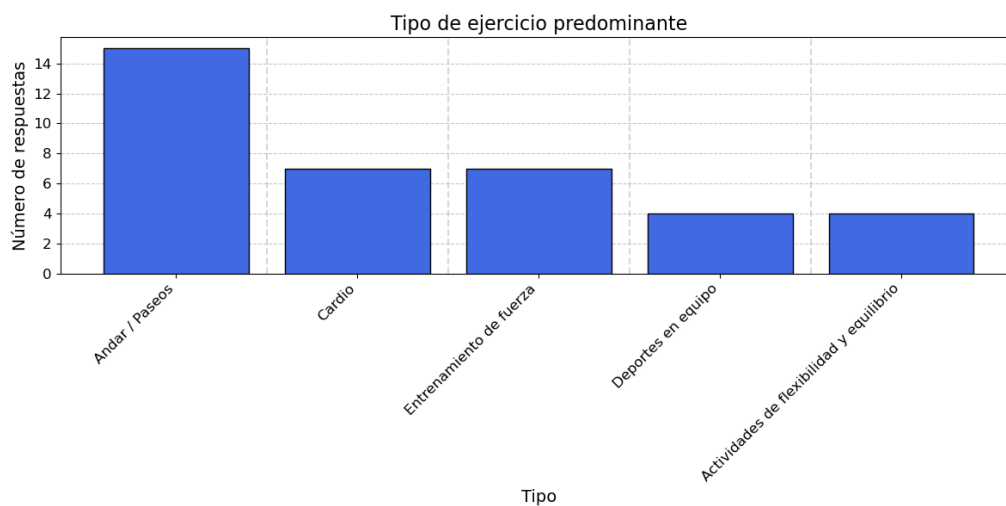


Figura 3.35: Tipo de ejercicio realizado por los usuarios tras la semana de pruebas

En cuanto a los lugares en los que se lleva a cabo estas actividades, se observa que no ha habido cambios significativos. Se mantiene el uso de espacios interiores como gimnasios o centros deportivos, aunque con una leve disminución en la cantidad de usuarios que lo realizan aquí. De esta misma manera, se ha observado un pequeño

incremento en el número de personas que realizan ejercicio al aire libre, sobre todo en zonas urbanas (Figura 3.36).

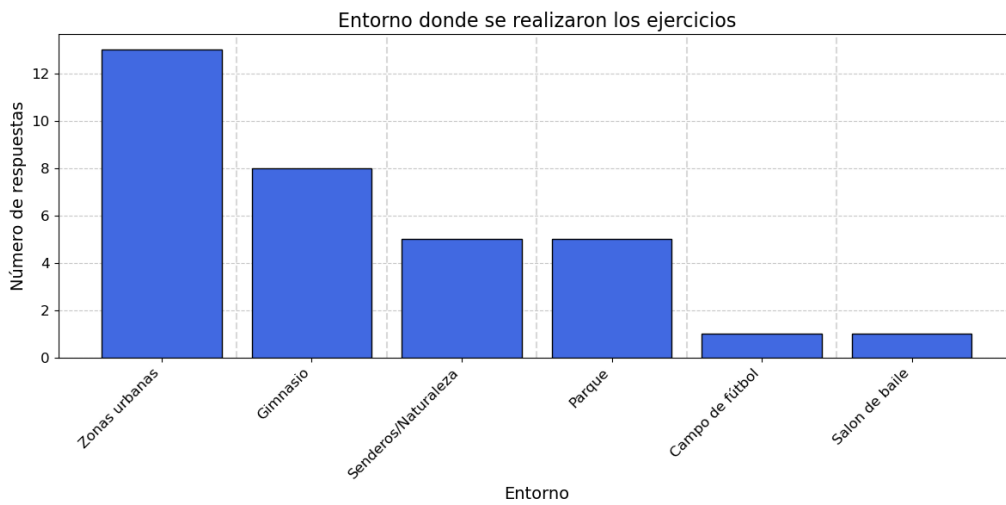


Figura 3.36: Dónde realizan el ejercicio los usuarios tras la semana de pruebas

En relación con el modo en que los usuarios realizan su actividad física, los datos evidencian una disminución en el número de participantes que hacen ejercicio en grupo. Mientras que las opciones de realizarlo de manera individual, con amigos o familia, o variando; mantienen un número de usuarios prácticamente idéntico (Figura 3.37).

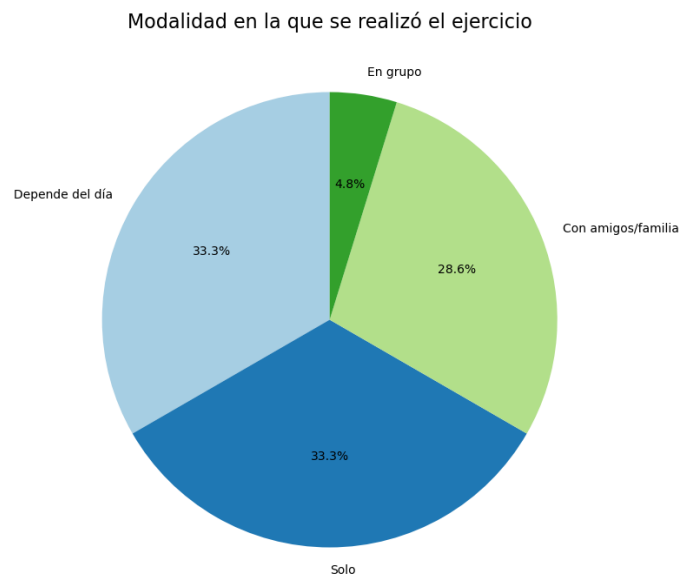


Figura 3.37: Cómo realizan el ejercicio los usuarios tras la semana de pruebas

Otro aspecto importante a destacar es el tiempo que los usuarios dedican a sus sesiones de ejercicio. Dados los resultados obtenidos, se mantiene la tendencia de

realizar sesiones de una duración superior a los 30 minutos, lo cual ya se había observado en el cuestionario inicial. Sin embargo, se nota un crecimiento en el grupo de usuarios que supera la hora de ejercicio por sesión. Casi la mitad de los encuestados realiza ahora más de 60 minutos de ejercicio, lo que indica un aumento en el tiempo total invertido en la actividad física durante la semana (Figura 3.38).

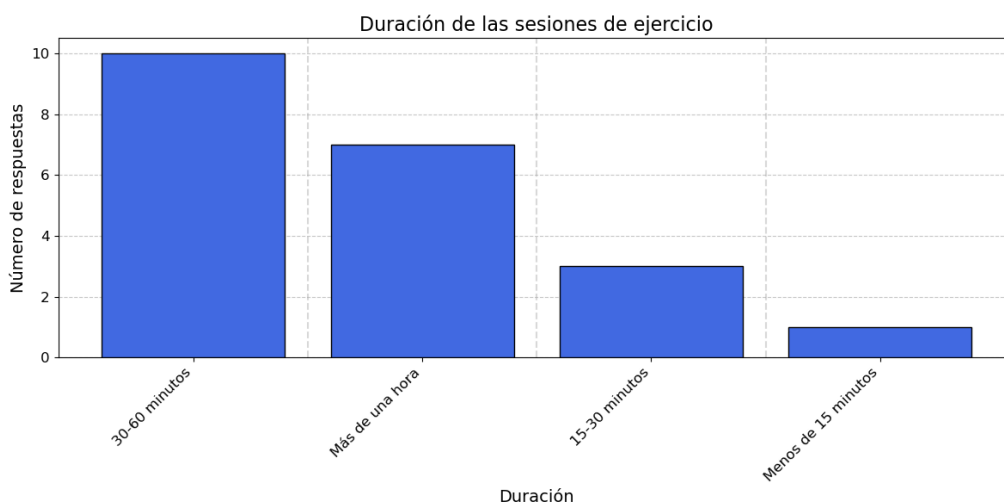


Figura 3.38: Cuánto tiempo le dedican al ejercicio los usuarios tras la semana de pruebas

Un punto relevante observado es el incremento considerable en el número de usuarios que hacen uso de tecnologías durante sus sesiones de ejercicio. Se puede observar un aumento en la cantidad de personas que recurren a dispositivos o aplicaciones para complementar sus ejercicios (Figura 3.39). Entre las tecnologías más utilizadas destacan las aplicaciones de rutas y mapas (Figura 3.40).

¿Se usaron apps o dispositivos durante el ejercicio?

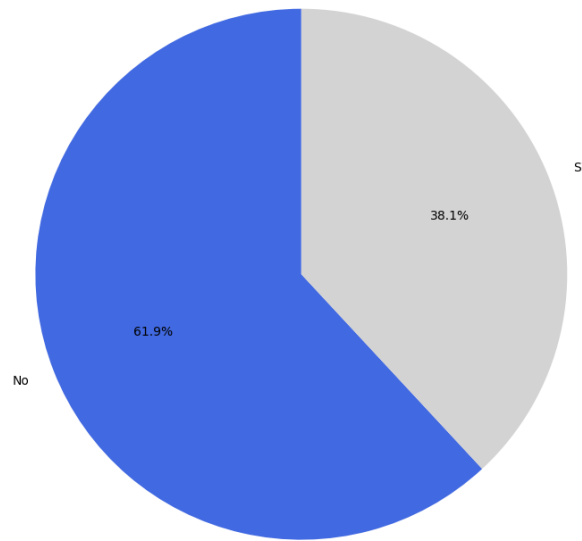


Figura 3.39: Usuarios que usan tecnologías durante los ejercicios tras la semana de pruebas

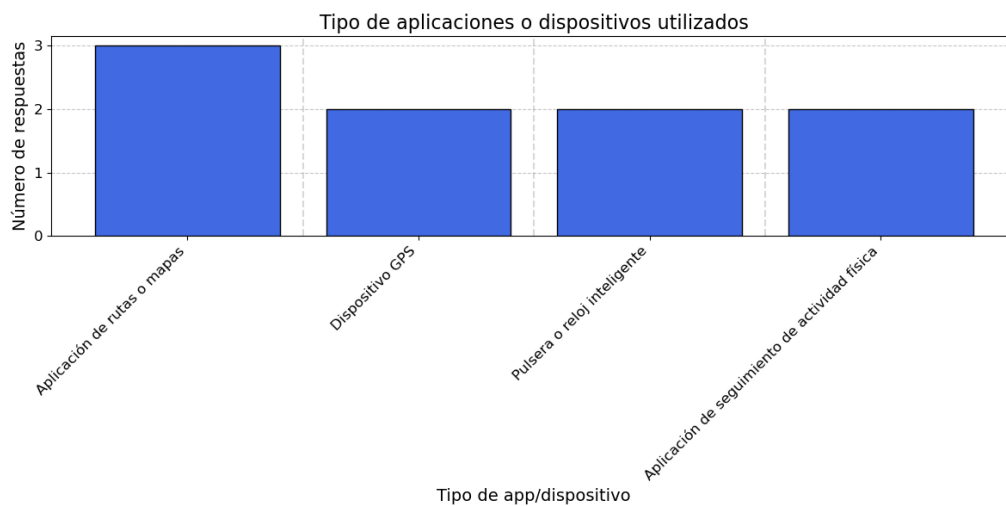


Figura 3.40: Tipos de tecnologías usadas tras la semana de pruebas

Es importante mencionar el uso de exergames por parte de los usuarios. Cerca de una cuarta parte de estos han incorporado o comenzado a incorporarlos dentro de sus rutinas de ejercicio (Figura 3.41). Cabe señalar que no se incluye el uso del juego de la prueba, *Critterons Capture*. Entre los títulos más mencionados destacan *Pokemon Go!* y *Orna*.

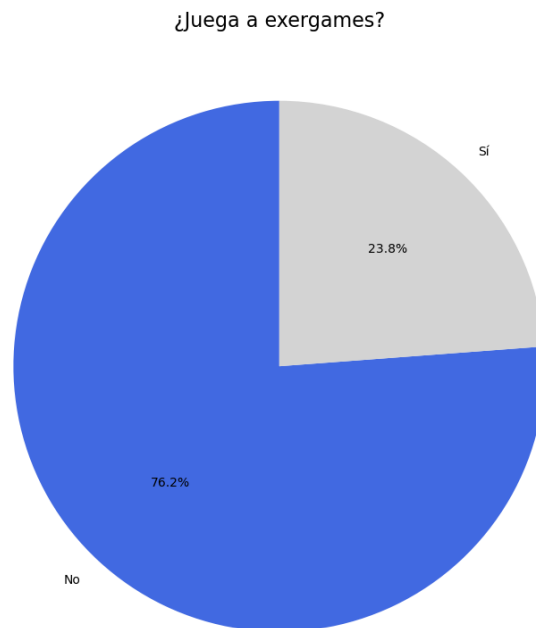


Figura 3.41: Usuarios que juegan exergames tras la semana de pruebas

Por último, tras la realización de este cuestionario, se mencionó a los usuarios que se habían incorporado nuevos critterons dentro del juego. De esta manera, podríamos ver el número de usuarios que volvían a usar el juego. Tras varios días de este anuncio, 4 de los 9 usuarios que realizaron las pruebas volvieron a utilizar la aplicación durante varios días.

3.6.5. Resultados de los cuestionarios diarios

Una vez estudiado el grupo de usuarios que jugaban al juego y su comportamiento durante la prueba pasamos a ver los resultados obtenidos por el otro grupo de usuarios. En este grupo, los usuarios no jugaban al videojuego, sino que completaron un cuestionario diario sobre el ejercicio que han realizado durante ese día durante los días de pruebas. Se recogieron datos los 7 primeros días. El número total de usuarios en este grupo es de 12 personas. Se observa cómo en los dos primeros días todos los usuarios realizaron ejercicio, aunque esto disminuyó durante los siguientes días. El día en el que menos ejercicio se hizo fue el quinto día (Figura 3.42). En cuanto al tipo de ejercicio se puede ver una clara tendencia a andar y realizar paseos durante todos los días (Figura 3.43). El tiempo que le han dedicado durante la semana, ha sido entre media hora y más de una hora, sobre todo durante los primeros días (Figura 3.44). Por último, el lugar donde se han realizado estos ejercicios ha sido habitualmente en zonas urbanas durante toda la semana, aunque también se realizaron en gimnasios y en la naturaleza de una manera general (Figura 3.45).



Figura 3.42: Número de usuarios que realizaron ejercicio por día

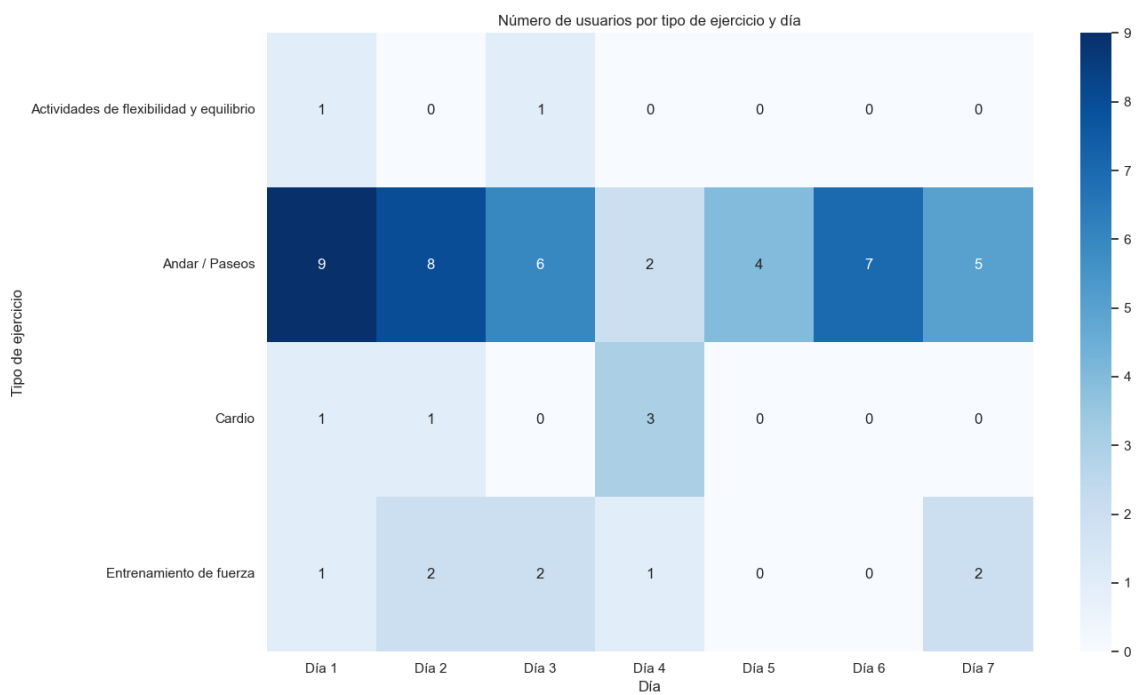


Figura 3.43: Qué tipo de ejercicio realizan los usuarios

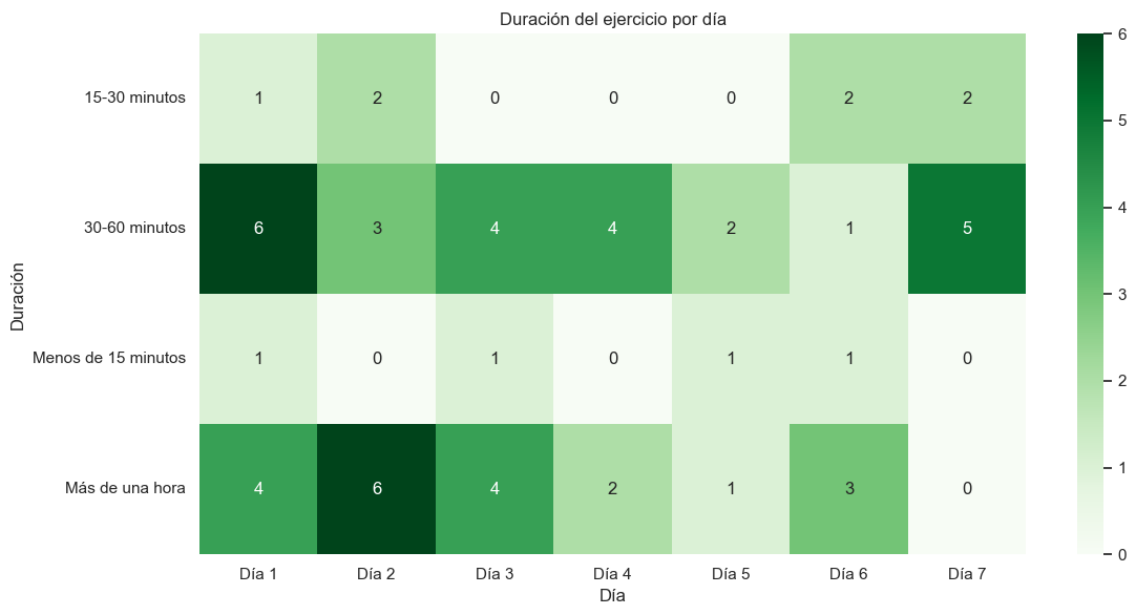


Figura 3.44: Cuánto ejercicio hacen los usuarios

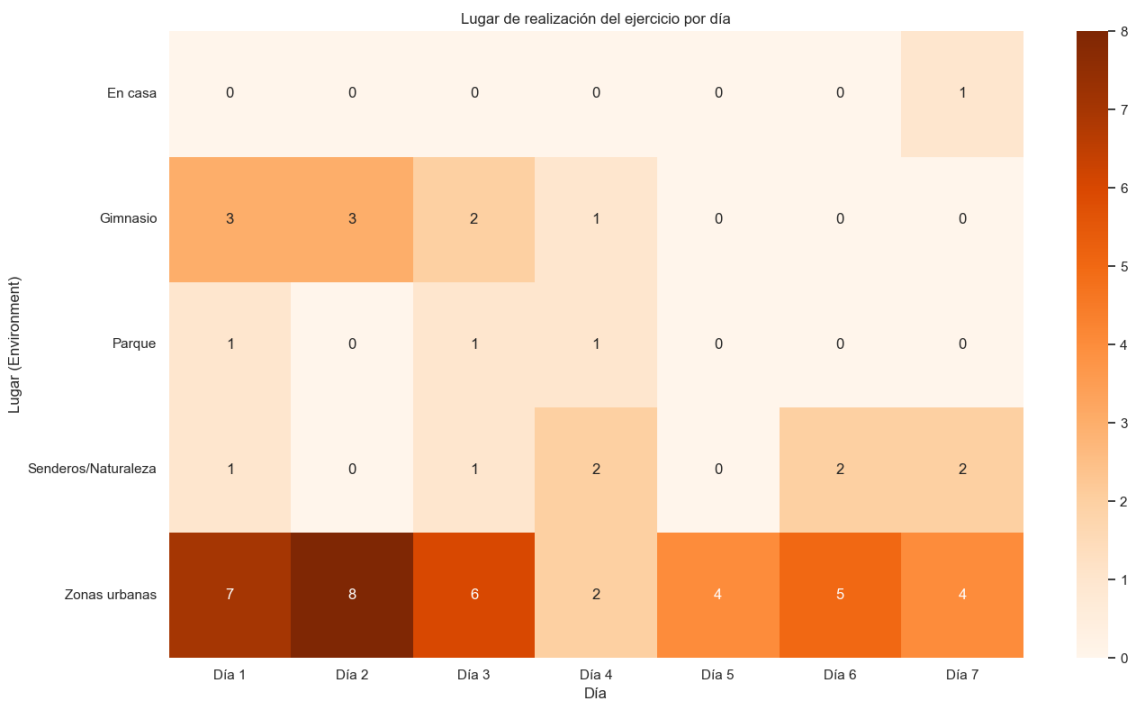


Figura 3.45: Dónde realizan ejercicio los usuarios

Cabe destacar, como se vio en el cuestionario inicial, que los dispositivos o aplicaciones siguen siendo algo que no todos los usuarios usan. Entre los usados están las aplicaciones de senderos y rutas y el uso de pulseras o relojes inteligentes (Figura 3.46).

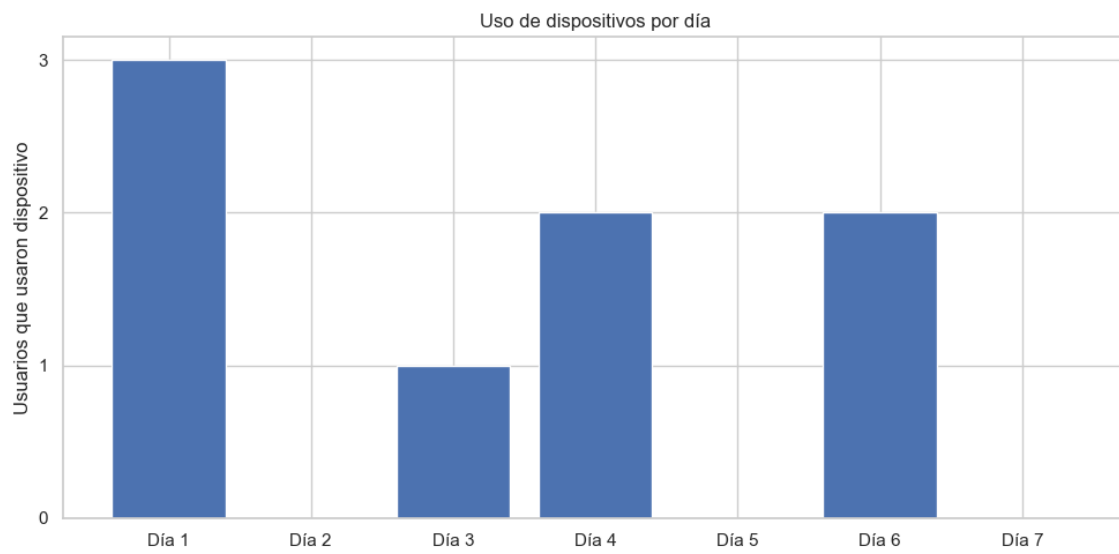


Figura 3.46: Uso de dispositivos/aplicaciones por día por los usuarios

Conclusiones y Trabajo Futuro

4.1. Conclusiones

El proyecto desarrollado ha dado como resultado un exergame y la realización de pruebas con usuarios. Gracias a esto, el juego ha quedado disponible para que otras personas puedan iterar sobre él y generar nuevas funcionalidades, tanto en el propio juego como en el servidor. Por ello, se han desarrollado los elementos y módulos de forma que fuese posible añadir nuevos componentes fácilmente. Para llevar esto a cabo, ha sido necesario realizar un estudio sobre el estado actual de los exergames, adquirir conocimientos sobre bases de datos y servidores, así como aprender el uso de nuevas herramientas y aplicaciones. Además, ha resultado clave el conocimiento previo sobre desarrollo de videojuegos de forma eficiente y estructurada.

En términos generales, el desarrollo del proyecto se ha llevado a cabo de forma continua y eficiente, cumpliendo con las tareas y respetando los tiempos establecidos. Esto ha permitido completar todos los apartados esenciales del proyecto. Del mismo modo, la buena coordinación entre los integrantes del equipo ha sido fundamental para que el trabajo se haya realizado de manera organizada, ayudándonos mutuamente en los aspectos necesarios.

Respecto a los resultados de las pruebas, se han podido observar distintas situaciones. En primer lugar, al comparar a los usuarios que han jugado con aquellos que no lo han hecho, se ha observado que los primeros han aumentado su número de pasos y su nivel de interacción durante los días festivos de la semana de prueba, mientras que en los segundos se ha producido el efecto contrario. Esto sugiere que los usuarios han utilizado el juego con fines lúdicos durante sus momentos de descanso, introduciendo sesiones de ejercicio en días en los que, como se ha visto con el otro grupo, no habría sido tan común hacerlo.

También, como se ha descrito, la interacción con el juego no ha variado tanto según la edad del usuario, sino más bien en función de cómo ha realizado su actividad

física. Las personas que ya caminaban con regularidad han incrementado su número de sesiones, mientras que otras lo han utilizado en entornos menos lúdicos, como durante el trabajo o al realizar otras actividades. Además, tras los días de prueba, los usuarios que han jugado al exergame han incrementado tanto el número de sesiones como el tiempo dedicado a ellas, en comparación con el mes previo a la prueba.

Otro dato relevante es que, una semana después de las pruebas, muchos de los usuarios han incorporado caminar o pasear como parte de su rutina de ejercicio. Asimismo, varios de los que han usado el juego han comenzado a integrar dispositivos electrónicos y aplicaciones similares, como otros exergames, en sus entrenamientos. En general, el juego ha sido bien recibido por los usuarios, aunque, como se ha recogido en las entrevistas, la mayoría ha coincidido en señalar la necesidad de un mayor número de objetivos y recompensas. También cabe destacar que no en todos los usuarios que han jugado al juego se ha tenido este mismo efecto. Las principales causas han sido la dificultad de entender el juego en un principio y la falta de incentivos.

Sin embargo, debido a la corta duración del juego, el cual ha sido diseñado para poder completarse en una semana si se juega de forma constante y al tiempo disponible para realizar las pruebas, no ha sido posible afirmar con certeza que se haya producido un efecto duradero en los usuarios. Para obtener conclusiones más sólidas, será necesario prolongar la interacción de los usuarios con el juego, ampliar la muestra de participantes y procurar que estos no tengan relación directa con los integrantes del proyecto. Solo así se podrá comprobar si los efectos observados se mantienen en el tiempo o, por el contrario, desaparecen.

Aunque no se puede afirmar con certeza que afecte de una manera positiva a largo plazo, a corto plazo se observa un aumento del ejercicio en los usuarios y un mayor interés en este tipo de juegos. También ha servido para que muchos usuarios que nunca habían probado estas experiencias tengan un primer contacto que ha sido percibido de una manera positiva.

4.2. Trabajo futuro

Tras el desarrollo, hemos logrado un juego sólido, plenamente funcional y que cumple con los diferentes objetivos que nos hemos propuesto durante el desarrollo. Sin embargo, sabemos que contamos con un gran margen de mejora que podríamos desarrollar en momentos futuros para pulir y expandir la experiencia.

En primer lugar, un punto importante y relativamente sencillo sobre el que trabajar sería mejorar el feedback que da la aplicación. Algunas mejoras que entrarían dentro de esta parte son:

- Incluir animaciones más vistosas para diferentes elementos de la aplicación.

- Rediseñar partes estéticas de la interfaz para que sean más llamativas y buscando un diseño más homogéneo.
- Optimizar la navegación dentro de la app, permitiendo que la experiencia sea más fluida.

Otro punto sobre el que trabajar sería el sistema de combate, en el cual se podrían mejorar algunos elementos como:

- Mejorar los controles del usuario, haciendo que devuelvan más feedback y sean más claros.
- Ampliar la cantidad y mejorar la calidad de los efectos, así como añadir variedad tanto en los fondos como en las acciones de los *critterons*.
- Implementar animaciones de inicio y final de combate más pulidas.

Otra posible mejora sería rediseñar diferentes dinámicas dentro del juego. Las paradas necesitan proporcionar otras recompensas o tener otra función para que el usuario busque más activamente interactuar con ellas. Los combates son reducidos en toma de decisiones, lo que puede llevar a desmotivar a algunos jugadores, por lo que añadir algún sistema de tipos a los ataques o a los propios *critterons* para aumentar la profundidad de los combates enriquecería estos en gran medida. El sistema de progreso es adecuado, pero proponer nuevas herramientas para obtener recompensas y progresar, como recorrer distancias en tiempos concretos o seguir rutas específicas, daría más variedad al ciclo de juego principal.

El factor social, aunque válido, consideramos que puede lograr resultados mucho mejores haciendo que los usuarios interactúen más con esta parte de la aplicación y ampliando así la vida útil del juego. En este sentido se podría:

- Permitir visitar los hoteles y ver más información sobre tus amigos.
- Rediseñar los rankings para hacerlos más llamativos y que no solo esté presente el de los pasos.
- Dar mayores recompensas por los combates con y contra amigos.
- Incluir visualmente el *critteron* de tu compañero si estás usando la opción de caminar junto con él.

Otras opciones como personalización del avatar o un sistema de combate entre jugadores que no se hayan agregado como amigos serían más costosas, pero encajaría con nuestra propuesta.

Agregar un mayor volumen de contenido, ya sea en habitaciones, *critterons*, zonas de combate, paradas, etc., lograría que los jugadores permanecieran más tiempo

jugando, ya que muchos han abandonado el juego cuando han sentido que han logrado obtener todo o todo lo que les interesaba. Unido a este punto, pero más complejo, sería la automatización de la generación de paradas para no limitar la zona geográfica donde jugar al juego, y la creación de eventos temporales o temáticos, así como recompensas que sirvieran de incentivo para que los jugadores volvieran al juego cada cierto tiempo.

Gran parte de estas mejoras potenciales las hemos reconocido gracias a las pruebas que hemos realizado con usuarios. Por lo que realizar más pruebas con usuarios, a ser posible con un mayor volumen de estos en cada una, sería fundamental para poder ir puliendo y mejorando la experiencia que ofrecemos. Estas pruebas podrían realizarse simultáneamente con las otras mejoras, permitiéndonos no solo detectar nuevos cambios a realizar, sino logrando retroalimentación inmediata de la eficacia de los cambios que hemos implementado.

Unido a esto, una mejora en el sistema de recogida de datos, ya sea con Xasu o con otra tecnología, nos permitiría poder medir mucho mejor cómo interactúa el usuario con la app y detectar los puntos que requieren ser mejorados; principalmente lograr que la información sea enviada a través de la red automáticamente y no haya que solicitar manualmente al usuario, y aumentar el volumen de eventos que registramos para poder hacer un mapa mucho más preciso de cada sesión.

Contamos con una buena base y multitud de campos para mejorar. Por lo que, con suficiente tiempo y un buen plan de desarrollo que intercale la implementación de mejoras con pruebas con usuarios, podemos obtener un producto atractivo que ofrezca una experiencia mucho más amplia.

Introduction

“The most important step a man can take. It’s not the first one, is it? It’s the next one. Always the next step, Dalinar”
— Brandon Sanderson

4.3. Motivation

The main motivation for doing this project is the opportunity to apply all the knowledge we have learned during the degree. In this way, the chance to create an online video game, as well as the development of the data structures and the necessary projects to launch it, is a very interesting challenge that we wanted to take on from the beginning. Also, this project not only lets us use what we already know, but it also allows us to explore tools and areas we had not worked with before, which makes the project even more interesting. Some of these include creating our own API, which is a set of calls and protocols that allow communication with data, and also using databases, among others.

On the other hand, the purpose of the game is to encourage healthy habits in players, which gives the project even more meaning. This gives us the chance to explore a field that we have been very interested in from the start.

Finally, doing this project helps us personally to learn and show our knowledge in a clear way, by creating a real and functional product, which is very important for our future careers.

4.4. Objectives

The main goal of this project is to create a video game — more specifically, an exergame. As we will explain in more detail later, an exergame is a game where body movements and physical activity are key to playing. Another important goal

is to check if the game can help improve players' healthy habits. To achieve this, we need to complete several smaller goals:

1. Study the current state of the exergames field. This way, we can better understand what already exists and which techniques are used to motivate players to do more physical activity.
2. Create an initial design of the video game. This also includes a study to choose the development environment that fits the needs of the project.
3. Study what information will be stored in the database and how it will be represented.
4. Develop the game after understanding the full idea and scope of the project.
5. Create an API to send and receive data, ensuring smooth information flow.
6. Once the API works in a local environment, connect it to the game, which should already be mostly developed at this point.
7. Deploy the API to make the game work outside of the local environment. Also, automate and document the steps needed to launch the game.
8. Design and carry out a test plan to collect user data in a controlled way.
9. Analyze the data collected during the tests.
10. Improve the game and identify future work based on the results.

4.5. Work Plan

As mentioned before, since there are many goals to complete, organizing the work is very important to finish the project on time and with the best quality.

To do this, we planned meetings every two weeks with the project tutors. In these meetings, we planned the work, checked progress, and solved doubts. Also, the work was divided into blocks that match the partial objectives.

Study and Initial Design

The goal of the first block was to set the foundation for the project. We started by looking at exergames and apps that use gamification to promote physical activity. We also looked for research articles to see how exergames are used in scientific studies. We saved the most important information in a document. After that, we created a Game Design Document with the key points of the game. Then, we

studied different development environments to find the one that best fits the project needs. Since it is an online game, one of the main challenges was how to store and access resources in a stable and efficient way. For this, we studied what kind of data we needed and how to represent it. Then, we designed the API to access these resources in the best way and selected the most efficient environment to develop it. We planned one month to complete this block.

Game Development

The goal of this block was to implement the game and create the API infrastructure. We did tests to check if the API and database were viable. After completing the tests, we developed both. The main focus was the connection between the API and the database, security, and creating the needed endpoints. These endpoints are the calls that allow interaction with the server. As we created new features, we added them to the game to test them. Once the systems were working, we worked on automating the API deployment and making it easy to launch in different environments. We developed one project to automate data uploading and another to allow an easy and guided deployment. This block took six months.

User Testing

In this block, we designed the experiment, which included the test script and the different questionnaires that the users had to answer. We also supported and guided the users during the tests and collected data from them to process later. This phase took one month.

Data Analysis

This was the final block of the project. After finishing the tests and collecting all the data, we processed the information to get accurate results. These results helped us to draw conclusions and evaluate the success of the project. This phase also took one month.

After completing all these blocks, the project was considered finished. For version control and project management, we used GitHub. The project repository is available at the following link: <https://github.com/RaulSaavedraRiera/TFG2024-AntonioPRaulS>

Conclusions and Future Work

4.6. Conclusions

The project has resulted in the creation of an exergame and a user testing plan. Thanks to this, the game is now available for others to build upon, adding new features to both the game and the server. For that reason, the components and modules were developed in a way that makes it easy to add new elements. To achieve this, we had to study the current state of exergames, learn about databases and servers, and get familiar with new tools and applications. Previous knowledge about efficient and structured game development was also very helpful.

Overall, the development of the project was steady and efficient, completing tasks and meeting deadlines set by our tutors and ourselves. This allowed us to finish all the essential parts of the project. Also, good coordination between team members was key to keeping the work organized and supporting each other when needed.

Regarding the results of the tests, we observed several situations. First, when comparing users who played the game with those who didn't, we noticed that the players increased their step count and interacted more during the holiday days of the testing week. In contrast, the non-players showed the opposite trend. This suggests that users played the game for fun during their free time, adding physical activity to days when they usually wouldn't.

Also, as described, interaction with the game didn't depend much on the user's age, but more on how they usually exercised. People who already walked regularly increased their sessions, while others used the game in more casual settings like at work or during other activities. In general, users who played the exergame had more sessions and spent more time exercising during the testing week compared to the previous month.

Another interesting result is that, a week after the testing ended, many users had added walking as part of their exercise routine. Some of those who played the game

also started using other similar devices and apps, like other exergames. Overall, the game was well received, although, based on the interviews, most users said there should be more goals and rewards. It's also important to note that not all players had the same positive results. The main issues were difficulty understanding the game at first and lack of motivation.

However, because the game is short (designed to be finished in a week of regular play), the user group mainly consisted of friends and family, and the testing time was limited, we cannot say for sure that the game has a lasting effect. To get stronger conclusions, we need to let users play for a longer time, increase the number of participants, and ensure they are not directly connected to the project team. Only then can we check if the effects last or fade away.

Even though we cannot confirm a long-term positive effect, in the short term, and considering the points above, we did see an increase in physical activity and user interest in this type of game. It also gave many users their first experience with these kinds of apps, and based on the data and interviews, it was mostly a positive one.

4.7. Future Work

We have created a solid and fully functional game that meets the goals we set. However, there's still a lot of room for improvement and expansion in the future to make the experience even better.

One relatively simple area to improve is the feedback given by the app. Some possible improvements are:

- Better animations for different elements in the app.
- Redesign parts of the interface to make it more attractive and consistent.
- Improve navigation to make the experience smoother.

Another key area is the combat system, which has a lot of potential for improvement. We can make battles look and feel much better by working on:

- Making the user controls clearer and giving more feedback.
- Adding more and better effects, and more variety in backgrounds and critteron actions.
- Creating more polished start and end animations for battles.

We should also redesign some game mechanics. The checkpoints need to give different rewards or have another purpose to encourage players to interact with them more. Battles are too simple and offer few choices, which can be boring. Adding a system of types (for attacks or critterons) could make the combat more interesting. The progress system works fine, but adding new ways to earn rewards, like walking specific distances in a set time or following routes, would add more variety to the core game loop.

The social part of the game works but could be much better if we encourage more interaction between users, which would also help keep the game interesting for longer. Ideas include:

- Let users visit hotels and see more info about their friends.
- Redesign the rankings to make them more engaging and not only based on step count.
- Give better rewards for battles with or against friends.
- Show your friend's critteron on screen when walking together.

These improvements are relatively easy to implement and would enhance the experience. Other features like avatar customization or a battle system for players who are not friends would take more work but would fit well with the game.

Adding more content—new rooms, critterons, battle zones, checkpoints, etc.—would help keep players interested. Many users stopped playing when they felt they had seen everything. A more complex improvement would be automatically generating checkpoints to avoid limiting the game to a small area. Also, adding temporary or themed events and time-limited rewards would give players reasons to return to the game regularly.

We identified most of these possible improvements through user testing. Doing more tests with a larger number of users is very important to keep improving the experience. These tests can be done while implementing other features, helping us get quick feedback on what works and what doesn't.

Along with that, improving the data collection system (using Xasu or other tools) would help us better understand how users interact with the app and find weak spots. For example, making the app send data automatically over the internet instead of manually, and tracking more events, would help create a much clearer picture of each session.

We have a solid foundation and many ways to improve. With enough time and a good development plan that includes both improvements and user testing, we can create a much more complete and engaging product.

Capítulo 5

Contribuciones Personales

Antonio Povedano Ortiz

Al inicio del proyecto, mi tarea principal consistió en investigar sobre diferentes exergames. Para ello, busqué los principales exergames actuales dentro del sector de los dispositivos móviles, exergames de consolas y exergames diseñados para estudiar su funcionamiento y su efecto en los jugadores. Con esta información, redacté un documento para tenerla de manera clara y poder usarla como referencia en los siguientes pasos del proyecto. Me centré especialmente en analizar las distintas formas que utilizaban para incentivar la actividad física entre los jugadores y, en el caso de los estudios, en cómo realizaron sus pruebas. Después, ambos integrantes investigamos la posibilidad de incluir la pulsera EmotiBit en el juego. Para ello, descargué los programas necesarios para probarla de manera local. Sin embargo, como ya se ha explicado, concluimos que no se podía integrar de manera efectiva en el proyecto de forma nativa.

Una vez completamos la investigación inicial y pusimos en común la información obtenida, comenzamos a diseñar el juego. Para ello, realizamos un GDD que fuimos completando de manera conjunta hasta tener una idea clara del videojuego que queríamos desarrollar.

Tras esto, mi trabajo consistió en investigar la estructura de los datos que necesitábamos para el juego, basándome en lo desarrollado en el GDD. De nuevo, de forma conjunta, desarrollamos pequeños proyectos de prueba para comprobar la viabilidad y funcionamiento del framework SpringBoot y de la base de datos MongoDB. Al confirmar su viabilidad, me encargué del desarrollo de la API, de los diferentes módulos necesarios para permitir la comunicación con los distintos elementos y estructuras del juego. Desarrollé así las llamadas HTTP correspondientes, así como la comunicación interna entre ellas, incluyendo la creación automática de las estructuras principales del juego, como GameInfo o UserInfo, y el volcado automático de

datos desde otras colecciones a estas estructuras. De esta manera, se logró la comunicación efectiva entre la API y la base de datos MongoDB, comprobándolo a través de Postman. Tras esto comencé a integrar las llamadas en el juego a través de las clases principales encargadas del flujo de datos. Primero, se creó la clase `ServerConnection`, que contenía de manera general todas las llamadas disponibles en la API. Posteriormente, desarrollé clases auxiliares para distribuir el trabajo de forma más organizada. Una vez estuvieron implementadas las llamadas, procedí a refactorizar el código para que dejara de obtener los datos de manera local y comenzara a comunicarse directamente con la base de datos. Durante esta etapa, para comprobar toda la funcionalidad, lanzaba el servidor de manera local. Durante este desarrollo, de forma conjunta creamos un módulo de seguridad que se encargaba de que las llamadas requirieran un token JWT en su encabezado, evitando así que cualquier usuario pudiera acceder a ellas.

Parte de mi trabajo también consistió en el desarrollo de funcionalidades del juego. Una gran parte fue la limpieza de código, refactorizaciones y revisiones periódicas de todo el bucle jugable. Además, trabajé en funcionalidades como el sistema de amigos, cuyo desarrollo estuvo estrechamente relacionado con el funcionamiento interno de la API, la curación de los critterons en base al tiempo (tanto mientras el jugador juega como cuando tiene el juego apagado), y la creación de pantallas dentro del juego.

También desarrollé un pequeño programa para la subida de datos estáticos del juego, como los critterons, las habitaciones o las marcas. Esto facilitó el despliegue de la base de datos, tanto de manera local como externa, haciendo esta tarea más cómoda y eficiente.

Como mencioné, parte de mi trabajo inicial fue estudiar cómo otros estudios realizaban sus pruebas. Por ello, cuando ya teníamos un bucle jugable estable, elaboré un plan de pruebas para comprobar el funcionamiento y los efectos del juego en los jugadores. Tras desarrollar este plan, creé los recursos necesarios, como cuestionarios y documentos informativos. Además, desarrollé el envío automático de correos electrónicos a los usuarios, ya que la cantidad de correos era alta y se enviaban diariamente a distintos usuarios. Durante la semana de pruebas, realicé actualizaciones para corregir errores dentro del juego y me aseguré de que las pruebas se llevaran a cabo correctamente, prestando ayuda a los usuarios cuando fue necesario.

Tras la finalización de las pruebas, recopilé los datos y los almacené en archivos `.json` para poder procesarlos de forma más sencilla. Con esta información, generé gráficos tanto de los datos iniciales de los usuarios como de los datos recogidos diariamente desde la base de datos, los datos obtenidos a través de los correos electrónicos diarios y los resultados de los cuestionarios finales.

Al igual que mi compañero, también dediqué tiempo a realizar este documento mostrando el trabajo realizado, redactando diversas secciones de la memoria que abarcan desde explicaciones iniciales y estado del arte hasta detalles del diseño del videojuego y análisis de los datos obtenidos.

Raúl Saavedra de la Riera

Al empezar el proyecto, mi tarea inicial consistió en buscar cuál era el entorno de desarrollo más adecuado para elaborar el proyecto. Con este objetivo, busqué los entornos de desarrollo más populares para el desarrollo móvil, tanto generalistas como especializados en videojuegos. Una vez tuve un número que consideré adecuado, obtuve los principales puntos fuertes y débiles de cada uno y comencé un proceso de selección donde fui, a través de varias fases y progresivamente, descartando diferentes opciones que o no cumplían con nuestras necesidades, o requerían desarrollo extra, hasta, entre los finalistas, seleccionar el motor de videojuegos Unity como tecnología sobre la que desarrollar el cliente del juego. También investigué junto a mi compañero el incluir la pulsera Emotibit en la aplicación, colaborando en las diferentes pruebas que realizamos, investigación que, como ya hemos visto, concluyó con la no integración de esta en el juego.

Una vez completada la fase de investigación y junto a mi compañero, desarrollamos un GDD donde definimos las diferentes características del videojuego que nos sirviera de guía para el resto del desarrollo. Una vez tuvimos definido el videojuego a desarrollar, reuní los diferentes paquetes de *assets* que forman parte del juego.

En las fases iniciales de desarrollo de la API colaboré junto a mi compañero con elementos como la autenticación o la gestión de algunas llamadas.

Mi siguiente tarea, y la que más volumen de trabajo me ha llevado, ha sido la de desarrollar gran parte del cliente en Unity. Mi compañero trabajó en aquellas partes más relacionadas con la conexión al servidor, como la parte social y funcionalidades sociales, entre otros. Por mi parte, desarrollé en primer lugar toda la lógica de navegación que conecta las diferentes partes del juego, así como probarlo para asegurar que el ciclo de juego se podía realizar correctamente. Tras esto, definí e implementé toda la sección del hotel dentro del juego, tanto las habitaciones como los objetos que representan a los *critterons* en este, así como los diferentes sistemas que dotan de lógica a esta parte del videojuego. Una vez el hotel estuvo completado, desarrollé el sistema de combate, creando de nuevo los objetos que representan los *critterons*, así como toda la funcionalidad para que estos pudieran llevarse a cabo. Implementé, a nivel visual e interactivo, las diferentes pestañas que conforman la parte social y de perfil del jugador. Para esta sección y todas las anteriores también creé los diferentes componentes que forman la interfaz de usuario, tanto aquellos interactivables como los que no, buscando seguir estructuras reutilizables para poder reusarlos en diferentes áreas del juego y simplificar su modificación.

Una vez el grueso de la aplicación estaba construido, la siguiente tarea fue la integración del movimiento en la aplicación. El primer elemento fue el podómetro, para el que tuve que realizar una pequeña investigación a través de internet de los diferentes enfoques para implementarlo; tras esto desarrollé varios prototipos sobre distintas aproximaciones hasta lograr tener la funcionalidad deseada, siendo un contador de pasos funcional, aislado del resto de la aplicación y fácilmente configurable.

Tras completarlo, y para poder obtener una primera versión completa del cliente, integré el sistema de GPS en el videojuego perteneciente al *add-on* Online Maps. Tuve que aprender a usarlo y probar diferentes configuraciones hasta lograr un mapa que se adaptara a la estética que buscamos. Una vez ya presente en el proyecto, modifiqué algunas de sus funcionalidades y las amplié con código desarrollado por mí para poder incluir las paradas, su interacción, el avatar del *critteron* en el mapa y los controles sobre este.

Con una primera versión del videojuego ya completa, mi trabajo hasta el inicio de las pruebas fue principalmente iterar sobre la aplicación para ir obteniendo un producto cada vez más pulido. Arreglar diferentes *bugs* que impedían el desarrollo del juego, modificar elementos de la UI o de algunas de las pantallas que no funcionaban adecuadamente o con un resultado por debajo de lo esperado, ampliar funcionalidad de algunos elementos, agregar sonido o refactorizar múltiples sistemas para lograr mayor robustez en la lógica del juego son varias de las tareas que realicé durante ese tiempo.

Durante las pruebas, me encargué principalmente del soporte de la aplicación, arreglando fallos menores que habían pasado desapercibidos y modificando partes que, gracias al *feedback* de los usuarios, concluimos que no funcionaban tal y como esperábamos. Tras la finalización de las pruebas, seguí trabajando en corregir errores y realizar mejoras menores para poder obtener una versión final más estable y cercana al objetivo que tuvimos en mente inicialmente.

Junto a mi compañero, he dedicado tiempo también a desarrollar este documento, mostrando el trabajo que hemos realizado en diferentes secciones como el estado de la cuestión, la descripción del trabajo o trabajos futuros.

Bibliografía

"They don't know I have my Spike lvl 12"

Jugador de Critteron Capture

ASOCIACIÓN ESPAÑOLA DE VIDEOJUEGOS (AEVI). La industria del videojuego en españa - anuario 2024. 2024. Accedido el 10 de mayo de 2025.

DECI, E. L., VALLERAND, R. J., PELLETIER, L. G. y RYAN, R. M. Motivation and education: The self-determination perspective. *Educational Psychologist*, vol. 26(3-4), páginas 325–346, 1991.

GAO, Z., CHEN, S., PASCO, D. y POPE, Z. A meta-analysis of active video games on health outcomes among children and adolescents. *Obesity Reviews*, vol. 16(9), páginas 783–794, 2015.

HARRIS, T. Georgia's got game: Why the gaming industry is larger than film, television and music combined. 2024. Accedido el 23 de enero de 2025.

HERRMANN, E., ENGELMANN, J. M. y TOMASELLO, M. Children engage in competitive altruism. *Journal of Experimental Child Psychology*, vol. 179, páginas 176–189, 2019. ISSN 0022-0965.

LAATO, S. Location-based games as exergames - from pokémon to the wizarding world. *International Journal of Serious Games*, vol. 7(1), página 79–95, 2020.

MARCELO COELHO, J. Z., SEAN FOLLMER. Pinizoro: Supporting the design of location-based games for children. En *Proceedings of the 9th International Conference on Interaction Design and Children (IDC '10)*, páginas 254–257. 2010.

SCARLE, S., DUNWELL, I., BASHFORD-ROGERS, T., SELMANOVIC, E., DEBATISTA, K., CHALMERS, A., POWELL, J. y ROBERTSON, W. Complete motion control of a serious game against obesity in children. En *2011 Third International Conference on Games and Virtual Worlds for Serious Applications*, páginas 115–120. 2011.

- SPINIFY. The history of gamification: How it all started. <https://spinify.com/blog/gamification-history/>, n.d. Accessed: 2025-05-15.
- STAIANO, A. E. y CALVERT, S. L. Exergames for physical education courses: Physical, social, and cognitive benefits. *Child Development Perspectives*, vol. 5(2), páginas 93–98, 2011.
- T. FUJIKI, M. P. Y. B. S. P. M. L., R. KAZAKOS. Fitness adventure: A mobile exergaming application. En *2009 IEEE International Symposium on Consumer Electronics*, páginas 1–4. 2009.
- WATTANASOONTORN, V., BOADA, I., GARCÍA, R. y SBERT, M. Serious games for health. *Entertainment Computing*, vol. 4(4), páginas 231–247, 2013. ISSN 1875-9521.
- WERBACH, K. y HUNTER, D. *For the Win: How Game Thinking can Revolutionize your Business*. 2012.
- YIM, M. y MOFFAT, A. C. Building heart rate into a mobile game. En *Proceedings of the 2009 ACM SIGGRAPH Symposium on Video Games*, páginas 125–129. 2009.

