



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escola Tècnica  
Superior d'Enginyeria  
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica  
Universitat Politècnica de València

# Desarrollo de un videojuego en Unity para el aprendizaje de matemáticas de tercero y cuarto curso de educación primaria

Trabajo Fin de Grado

**Grado en Ingeniería Informática**

**Autor:** Gabriel Caracava Gosp

**Tutor:** Ramón Pascual Mollá Vayá

[Curso 2020-2021]

Desarrollo de un videojuego en Unity para el aprendizaje de matemáticas de tercero y cuarto curso de educación primaria

# Agradecimientos

---

Me gustaría agradecer a mi tutor, Ramón Pascual Mollá, por sus consejos, apoyo y orientación durante la elaboración de este TFG.

Al profesor Patricio Letelier por sus clases y recomendaciones a la hora de abordar el desarrollo software.

A mi compañero Juan Girbés por sus ideas y consejos cuando los necesitaba.

A mis padres, por su apoyo tanto en los buenos como malos momentos durante la carrera.

Gracias a todos por brindarme la mano durante estos cuatro años.

# Resumen

---

Los videojuegos son actualmente uno de los entretenimientos más populares en los niños. Las matemáticas nos acompañan en nuestra vida cotidiana y son la base de la informática. Que mejor forma de aprender matemáticas que, desde pequeños, a través de un videojuego que nos entretiene.

Este proyecto tiene como objetivo mostrar todo el proceso de diseño y elaboración de un producto software de entretenimiento desde cero. Para ello se ha desarrollado un videojuego educativo donde se aplican los procedimientos necesarios para la elaboración de un producto software. El videojuego desarrollado es de carácter educativo y tiene como objetivo ayudar en el aprendizaje de las matemáticas de una forma entretenida y animada a niños de tercero y cuarto de primaria, es decir el segundo ciclo de la Educación Primaria. Los contenidos que abarca el videojuego son el sistema numérico, sumas, restas, multiplicaciones, divisiones, fracciones, números romanos entre otros. Para ello, se ha llevado a cabo un estudio en profundidad de las diferentes técnicas y metodología utilizadas para la enseñanza a estas edades, también se han comparado actuales aplicaciones/juegos educativos del mercado, y se han utilizado libros de primaria de la editorial Santillana para la obtención de ejercicios adecuados al nivel.

La plataforma de desarrollo a utilizar ha sido Unity, uno de los motores de videojuegos más populares del sector, ya que Unity ofrece una gran cantidad herramientas que han permitido desarrollar un producto completo.

**Palabras clave:** videojuego, Unity, matemáticas, educación, primaria.

# Abstract

---

Video games are currently one of the most popular entertainment for children. Mathematics accompany us in our daily lives and are the basis of computing. What better way to learn mathematics than from a young age through a video game that entertains us.

This project aims to show the entire process of designing and developing an entertainment software product from scratch. For this, an educational video game has been developed where the necessary procedures are applied for the development of a software product. The video game developed is educational in nature and aims to help children in the third and fourth years of primary education (second cycle of Primary Education) learn mathematics in an entertaining and animated way. The contents covered by the video game are the number system, addition, subtraction, multiplication, division, fractions, Roman numeral, among others. For this, an in-depth study of the different techniques and methodology used for teaching at these ages has been carried out, current educational applications / games on the market have also been compared, and elementary books from the Santillana publishing house have been used to obtain exercises appropriate to the level.

The development platform to be used has been Unity, one of the most popular video game engines in the sector, since Unity offers a large number of tools that have made it possible to develop a complete product.

**Keywords:** videogame, Unity, matemáticas, education, primary.

# Tabla de contenidos

1.	Introducción.....	9
1.1	Motivación .....	9
1.2	Objetivos .....	10
1.3	Estructura .....	11
2.	Estudio .....	12
2.1	Estado del arte .....	12
2.1.1	Historia de los videojuegos educativos .....	14
2.1.2	Videojuegos similares.....	15
2.1.3	Propuesta.....	20
2.2	Fundamentos teóricos .....	21
2.2.1	Teoría del Aprendizaje de Piaget.....	22
2.3	Técnicas utilizadas para la enseñanza a través de juegos.....	23
2.4	Técnicas aplicadas a Maths.....	25
2.5	Contenidos a implementar .....	27
2.3.1	Numeración .....	28
2.3.2	Cálculo mental y operaciones.....	30
3.	Análisis.....	33
3.1	Elicitación de requisitos.....	33
3.1.1	Glosario .....	34
3.1.2	Modelo de dominio.....	35
3.1.3	Diagrama de contexto.....	35
3.2	Identificación de requisitos .....	36
3.3	Especificación casos de uso .....	37
3.4	Metodología .....	41
3.5	Planificación .....	41
4.	Proceso de desarrollo .....	45
4.1	Arquitectura del videojuego.....	45
4.2	Patrones aplicados .....	46
4.2.1	Singleton.....	46
4.2.2	Observador .....	49
4.2.3	Mediador .....	52
4.3	Pruebas .....	54
4.3.1	Pruebas generales de rendimiento .....	54
4.3.2	Pruebas realizadas durante los sprints.....	57

4.3.3	Pruebas con niños.....	57
5.	Tecnologías utilizadas .....	64
5.1	Entorno de Desarrollo: Unity .....	64
5.2	Piskel.....	65
5.3	Genymodel y Creately.....	67
6.	Conclusiones .....	68
7.	Trabajo futuro .....	69
	Referencias y bibliografía .....	70
	Índice de ilustraciones.....	72
	Definiciones, abreviaturas y acrónimos .....	73
	Anexo I: documento de diseño .....	74





# 1. Introducción

---

Este documento contiene todo el proceso seguido para elaborar un producto software de entretenimiento, abarcando desde su idea inicial hasta la elaboración final del producto. Para ello se ha elaborado un videojuego de carácter educativo destinado al aprendizaje de matemáticas de tercero y cuarto curso de educación primaria.

Las matemáticas son la base de la informática y nos acompañan en nuestra vida cotidiana, que mejor forma de aprenderlas que desde pequeños a través de juegos. Con la llegada de las Tecnologías de la Información y de las Comunicaciones (TIC), cada vez, es más fácil ver como se incorporan a las nuevas metodologías de enseñanza (1). Así pues, existen aplicaciones que ayudan en la educación ofreciendo soporte en el aula, aportando realimentación, evaluando mediante rompecabezas, y que además aumentan la motivación y participación del alumnado como, por ejemplo, *Socrative*<sup>1</sup>. Por otro lado, también existen aplicaciones que directamente ofrecen contenidos para formar a sus usuarios como podría ser *Duolingo*<sup>2</sup>, la cual permite aprender idiomas con tan solo unos minutos al día mediante lecciones cortas. Diferentes aplicaciones con diferentes funcionalidades pero que tienen un objetivo común, la enseñanza.

Este trabajo de fin de grado se centra en un videojuego educativo de carácter matemático, destinado a alumnos de tercero y cuarto curso de primaria en España. No obstante, la idea de elegir estos dos cursos viene como objetivo de que la aplicación sea tanto útil para alumnos de quinto o sexto de primaria que quieran repasar contenidos de cursos anteriores, o para alumnos de primero o segundo de primaria que quieran ir un paso más allá y puedan practicar ejercicios de un nivel superior de una forma entretenida. Este trabajo de final de grado tiene como objetivo el aprendizaje basado en el juego (2), un modelo de aprendizaje que resulta motivar y facilita la participación del alumno en prácticas de aprendizaje activo (3).

## 1.1 Motivación

---

Cada vez es más frecuente ver la tecnología combinada con la enseñanza (4). El avance de las TIC en los últimos años ha hecho inviable no introducir la tecnología en el ámbito académico. Además, cada vez se elaboran más recursos educativos en formato digital, aumentado la variedad tanto en contenido educativo como en la metodología empleada para enseñar, así como su alcance al público. Es extraño no encontrar una editorial hoy en día que no cuente con recursos electrónicos de refuerzo para los libros de papel, webs de seguimiento y progreso, gestión de escuelas, etc. *Progrentis*<sup>3</sup>, por ejemplo, es una plataforma tecnológica de habilidades digitales que desarrolla el pensamiento creativo. Así pues, las TIC se convierten en el enlace de conexión entre el profesor y el alumno para el intercambio de información lo que deriva en la construcción conjunta de conocimientos (5). Entre los recursos tecnológicos utilizados en el ámbito académico

---

<sup>1</sup> <https://www.socrative.com/>

<sup>2</sup> <https://es.duolingo.com/>

<sup>3</sup> <https://www.progrentis.com/>

destacan: las páginas web, el correo electrónico, las bibliotecas en línea, las bases de datos, autoevaluaciones, redes sociales, blocs, videoconferencias, plataformas, como, por ejemplo, Microsoft Teams<sup>4</sup> (plataforma unificada de comunicación y colaboración que combina chat, videoconferencias, almacenamiento de archivos, entre otros) o RiuNet<sup>5</sup> (el Repositorio Institucional de la Universitat Politècnica de València).

La principal motivación para el desarrollo de este proyecto surge de las ganas de querer crear un producto software útil, que garantice una mejora de competencias matemáticas, mostrando todo lo aprendido durante estos cuatro años de carrera. La idea de desarrollar un videojuego de matemáticas nace del propio interés en las mismas y su importancia tanto en la informática como en nuestra vida cotidiana.

Desde pequeño me ha gustado crear cosas, siendo pequeño jugaba con imanes que se pegaban unos con otros y permitían recrear las estructuras de nuestra imaginación. Del paso de los imanes a los *Legos*, estos ya vienen con instrucciones y te permiten crear el producto específico para el que han sido diseñados, sin embargo, lo que realmente apasiona es utilizar las piezas para crear nuestros propios diseños. De los *Legos* pasamos a *Arduino*, esto ya no es un producto para tan niños pues requiere cierta complejidad, es aquí donde nace al interés por la informática. La informática, nos permite crear los contenidos que nuestra cabeza pueda imaginar, es un mundo con una inmensa cantidad de posibilidades y caminos aún no descubiertos. Por todo ello, surge el interés en la ingeniería informática, y una vez dentro, en la rama de ingeniería del software. Que mejor forma de ser creativo, que ponerse a desarrollar productos software de diferentes temáticas y con diferentes tecnologías.

La informática nos permite dar rienda suelta a nuestra imaginación.

## 1.2 Objetivos

---

El principal objetivo de este trabajo es mostrar todo el proceso de diseño y elaboración de un videojuego 2D para plataforma móvil y ordenador. El videojuego va destinado a enseñar matemáticas del segundo ciclo de Educación Primaria (tercero y cuarto curso de primaria). Para lograrlo, se han marcado los siguientes objetivos:

- Abordar todas las etapas del desarrollo software.
- Utilizar libros de texto de primaria para obtener ejercicios adecuados al nivel.
- Estudiar las metodologías educativas para enseñar a través de los videojuegos.
- Analizar otros videojuegos educativos similares.
- Crear un videojuego de matemáticas educativo.

La plataforma de desarrollo a utilizar ha sido Unity<sup>6</sup>, uno de los motores de videojuegos más populares del sector, ya que Unity ofrece una gran cantidad herramientas y facilidades que permiten hacer un producto completo. Para la creación de las imágenes

---

<sup>4</sup> <https://www.microsoft.com/es-es/microsoft-teams/group-chat-software>

<sup>5</sup> <https://riunet.upv.es/>

<sup>6</sup> <https://unity.com/es>

y *sprites*<sup>7</sup> se ha utilizado Piskel<sup>8</sup> una herramienta gratuita que permite crear ilustraciones y *sprites* de una forma sencilla pintando pixeles, si la necesidad de ser un diseñador profesional.

## 1.3 Estructura

---

Para finalizar el primer capítulo, se expondrá una lista con los capítulos y contenidos que forman el documento donde se pretender aclarar la estructura del mismo y resaltar los puntos más importantes de cada apartado.

- Capítulo 1: introduce la motivación, elección que lleva a realizar este trabajo fin de grado y los objetivos a alcanzar.
- Capítulo 2: en este capítulo se explica el estado actual del que se parte, la historia y videojuegos similares, posteriormente se realiza un estudio de los contenidos a implementar y de las técnicas utilizadas para enseñar.
- Capítulo 3: detalla el proceso de análisis que se ha realizado previamente al desarrollo de la aplicación, identificación de requisitos, diagramas, casos de uso, metodología a seguir para el desarrollo del proyecto, así como la planificación del mismo.
- Capítulo 4: se explica el proceso de desarrollo seguido, apoyándose de la arquitectura de la aplicación, patrones aplicados y pruebas realizadas para asegurar un correcto desarrollo software.
- Capítulo 5: se presentan las tecnologías utilizadas.
- Capítulo 6: se exponen las conclusiones finales obtenidas tras el desarrollo del videojuego y los resultados obtenidos, así como algunas propuestas de trabajos futuros sobre él.
- Capítulo 7: se presentan las referencias y bibliografía, el anexo de ilustraciones, el de definiciones, abreviaturas y acrónimos para aclarar conceptos del trabajo, y el documento de diseño del videojuego.

---

<sup>7</sup> Serie de imágenes unidas en un mismo archivo una al lado de otra y que representan al mismo personaje (u objeto) en distintas posiciones.

<sup>8</sup> <https://www.piskelapp.com/>



## 2. Estudio

---

### 2.1 Estado del arte

---

Es frecuente escuchar como ciertas personas consideran que el uso de los videojuegos tiene efectos adversos en la salud de sus jugadores (6). No obstante, diferentes estudios han probado que, aunque los videojuegos podrían perjudicar la salud como consecuencia de largas sesiones de juego sobrepasando tiempos razonables, también lo es que, estableciendo prácticas de juego como, por ejemplo, tiempo apropiado, sensatez y un buen entorno, dicha actividad puede resultar placentera y segura. (7)

El estado del arte engloba tres categorías. La primera tiene que ver con la comunidad educativa, en concreto la percepción positiva o negativa que profesores, estudiantes, padres y madres tienen acerca del uso de videojuegos. La segunda categoría expone experiencias en el aula sobre el uso de videojuegos y cuáles son los resultados en el proceso de aprendizaje obtenidos como resultados de las pruebas realizadas. Por último, la tercera se centra en el uso de videojuegos para la enseñanza de matemáticas.

La Asociación Española de Distribuidores y Editores de Software de Entretenimiento aDeSe llevó a cabo una investigación en 2012 en la que eligió objetivamente a profesores y padres de niños entre 5 y 12 años, de diversas localizaciones de España, con el propósito de instituir pruebas que avalasen la capacidad pedagógica del uso del videojuego como herramienta educativa y de desarrollo infantil. Entre los resultados obtenidos, destaca con un 94% el uso habitual o esporádico de dispositivos electrónicos con estudiantes por parte de sus profesores (ver figura 1). Uno de cada tres profesores afirma haber realizado tareas con videojuegos en el último año, y la mayoría hace uso de juegos educativos en asignaturas como conocimiento del medio, matemáticas, lengua española e inglés. El 79% del profesorado da el visto bueno al uso de los videojuegos como herramienta educativa en la enseñanza primaria, y los profesores opuestos a este uso (cerca del 20%) principalmente se oponen por no creer en la efectividad de los videojuegos (no se amoldan a las necesidades educativas) o por el desconocimiento de la herramienta. Los profesores participantes de la encuesta señalaron que los videojuegos resultan favorecedores para el desarrollo de las habilidades cognitivas (87,5%), psico-motoras (80%) y de las capacidades personales (64,2%). Por otro lado, no hay oposición por parte de los padres a la utilización de los videojuegos en clase, y el 92% acepta su uso como herramienta educativa. Por otra parte, el 80% de los estudiantes considera que el videojuego es un buen recurso y el 73% desearía que se utilizaran más recursos TIC en el aula. (8)

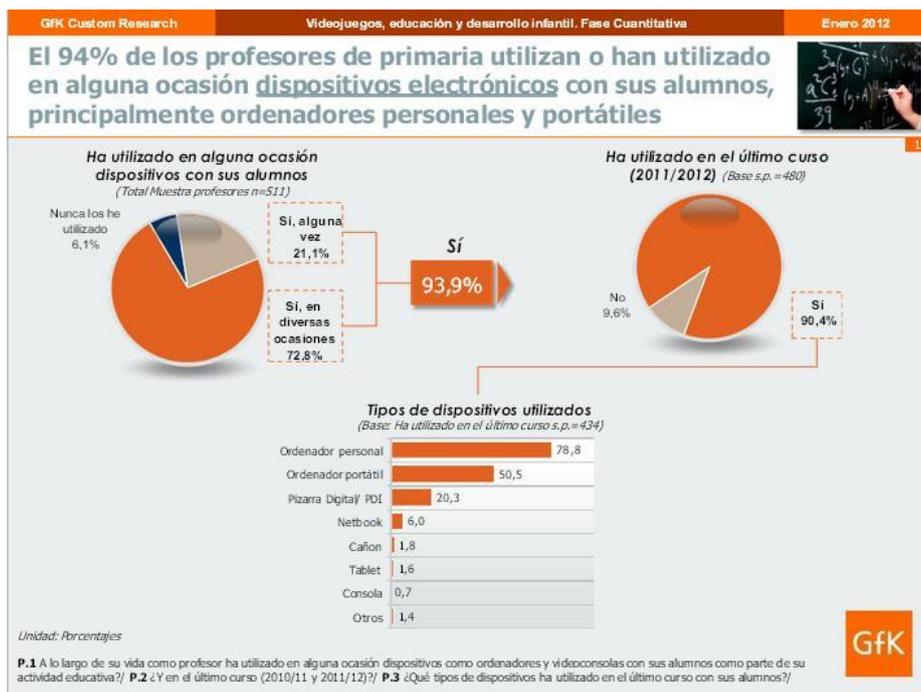


Figura 1: profesores que han utilizado videojuegos en la educación

La investigación de Begoña Gros (2007), muestra los frutos de su análisis sobre la relación de pares, el rol del docente y los aportes del videojuego. Para llevar a cabo la investigación se seleccionaron a 78 estudiantes de entre 12 y 14 años, quienes efectuaron tres ejemplos de pruebas sobre el uso del videojuego: indagación, nivelación y evaluación. Gracias a la metodología de estudio de caso de tipo descriptivo, se logró realizar una recopilación y estudio de información que posibilitó determinar las prácticas e interacciones manifestadas en la relación entre pares, el rol docente, y los aportes del videojuego.

Los resultados del estudio realizado por Begoña Gros pueden resumirse en diversos aspectos. El primero, relacionado con el tiempo y alude a la dificultad para ajustar el tiempo dedicado al juego con el tiempo asignado a una asignatura. Las secuencias de juego hay que planificarlas, especialmente frente al uso de juegos de estrategia cuya historia requiere de tiempo y afán para su avance. Otro aspecto se corresponde con los contenidos de los videojuegos, puesto que los resultados destacan la no necesaria similitud entre los juegos y asignaturas específicas. Pese al provecho del contenido para la tarea académica, los juegos presentan contenidos mucho más interdisciplinarios, siendo recomendable no fragmentar el aprendizaje a través de las diferentes disciplinas sino integrarlo de forma interdisciplinaria. El tercer aspecto, está relacionado con la inseguridad de los profesores al considerar a los alumnos mejores jugadores que ellos. Por ello, es preciso, hacer entender a los profesores que no deben competir con las habilidades sus alumnos. Pese a tener mayor conocimiento y experiencia en el uso de videojuegos, los niños carecen de la reflexión y los elementos críticos que el profesor puede aportar. El último punto está relacionado con el manejo de conceptos. En algunos juegos de simulación es necesario considerar los conceptos que se trabajan y saber contrastar los puntos de conexión con el conocimiento científico. (9)

Son diversas las fuentes, donde se presentan juegos como: Civilization para ilustrar conceptos de ciencias, historia o geografía; Minecraft para aprender a trabajar en equipo;

World of tanks para promover el liderazgo. Pero pese a existir miles de juegos de matemáticas, las fuentes no llegan a un consenso común sobre un determinado título. Posiblemente, fruto de la inmensa cantidad de títulos publicados es difícil establecer cuál de ellos es el mejor para determinado ámbito, edad o contexto. Son muy pocas las investigaciones localizadas en nuestro país sobre el uso del videojuego en el aula de matemáticas (10). A pesar de ello, en el trabajo de final de máster de Cristina Valcárcel (11) donde ya aborda el uso videojuego matemático en la Educación Secundaria, tras realizar el estudio concluye, que la utilización de videojuegos matemáticos en el aula favorece a los alumnos a que adquieran la competencia matemática necesaria mediante entornos de aprendizaje motivadores y significativos. (12)

## 2.1.1 Historia de los videojuegos educativos

---

La incorporación de mecánicas y dinámicas pertenecientes a juegos y videojuegos en entornos no lúdicos lleva practicándose desde hace tiempo, pero su fenómeno ha adquirido mayor importancia en estos últimos años.

Remontándose al inicio, en el año 1985 se lanza el videojuego educativo *Carmen Sandiego*<sup>9</sup> (basado en la serie *Carmen Sandiego*, de la cual se han hecho muchas versiones desde su lanzamiento), orientado a estudiantes de primaria y con el propósito de enseñar conceptos básicos de geografía. El videojuego trata de seguir las pistas de Carmen Sandiego, una ladrona de guante blanco. Para ello, el jugador debe recabar pistas e ir atando cabos a partir de un razonamiento lógico para saber a qué lugar viaja la ladrona con el fin de atraparla. A partir de títulos como este o similares como *Reader Rabbit*<sup>10</sup>, hicieron que los videojuegos educativos ganaran popularidad mundial incrementando el desarrollo de títulos educativos, no solamente los destinados a la población en general sino, cada día más, orientados a alumnos y cursos específicos. (13)

Haciendo un breve resumen de la historia, en los años 1990-95 aparecen novedosamente plataformas como las consolas portátiles y teléfonos móviles, haciendo los juegos cada día más sofisticados. Entre el 1996 y el 2000, llegan los primeros *mundos virtuales* donde los usuarios pueden crear contenido. A partir del 2001, empieza a hacerse popular el multijugador online y el mercado apuesta por consumidores no-tradicionales. Entre el 2006 y el 2010 Nintendo innova en el mercado de los videojuegos presentando la interacción física con los juegos a través de su consola *Wii*. A partir del 2011, los móviles y las tabletas pasan a ser las principales plataformas para jugar y los juegos comienzan a ser utilizados como una herramienta de aprendizaje en el sector de la educación transmitiendo conceptos a los alumnos, proceso que ha continuado hasta día de hoy.

En el 2020, la *EdTech* (tecnología educativa) ha adquirido un gran protagonismo debido a la pandemia mundial, que ha forzado a acelerar el proceso de digitalización de contenidos educativos apoyándose en la tecnología (14). Según *Statista*, las previsiones apuntan a que el mercado mundial del software y aplicaciones educativas presentarán una tendencia de fuerte crecimiento en los próximos años, en la figura 2 pueden observarse estas previsiones.

---

<sup>9</sup> <http://edu4.me/carmen-sandiego-un-videojuego-educativo-que-se-todavia-perdura/>

<sup>10</sup> [https://the-jh-movie-collection-official.fandom.com/wiki/Reader\\_Rabbit](https://the-jh-movie-collection-official.fandom.com/wiki/Reader_Rabbit)

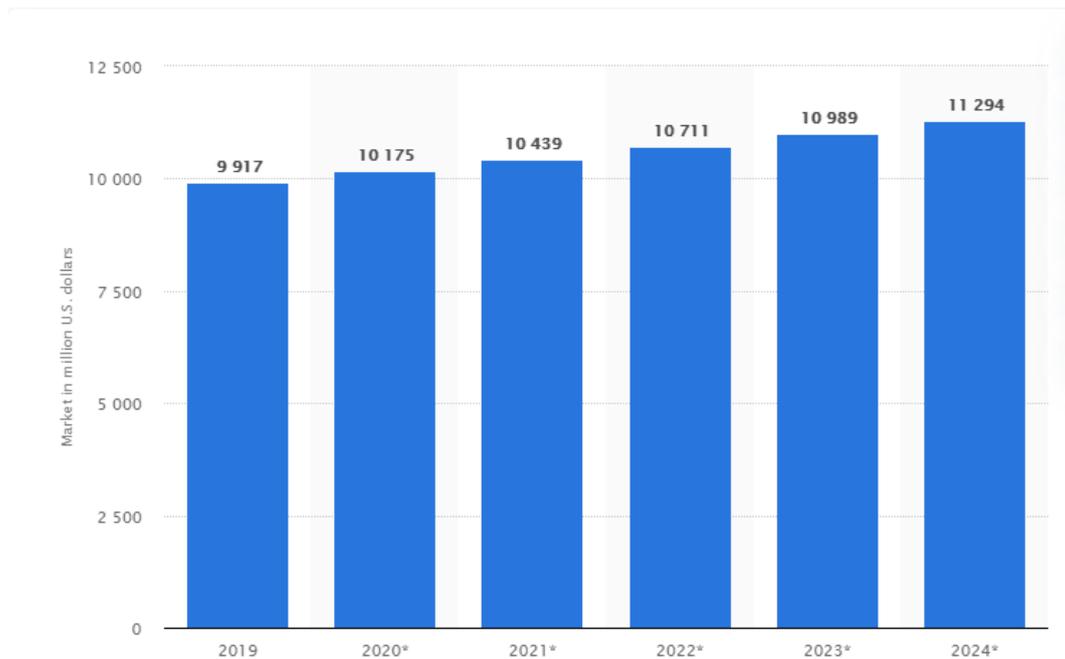


Figura 2: tamaño del mercado de software y aplicaciones educativas en todo el mundo de 2019 a 2024 (en millones de dólares estadounidenses), Statista

## 2.1.2 Videjuegos similares

Al inicio del proceso de desarrollo del prototipo de Maths, se realizó un estudio en profundidad de videjuegos similares para extraer ideas sobre lo que resultaba interesante o atractivo de estos juegos y que se podía añadir o mejorar para hacerlo destacar de los demás. Se han probado distintos juegos, pero algunos carecían de relevancia como para ser explicados en este documento. A continuación, se analizan los resultados de tres de los videjuegos probados:

### **Monster Numbers<sup>11</sup>:**

Es un videjuego disponible para plataforma móvil y computadora. Ha sido probado en la versión gratuita para Android, ya que también incluye una versión de pago cuya principal diferencia es que incluye más niveles de juego y no contiene publicidad. Cuenta con más de un millón de descargas para Android y una valoración por los usuarios de 4,2 sobre 5.

Monster Numbers es un videjuego educativo de matemáticas con sumas, restas, tablas de multiplicar y divisiones. Por una parte, cuenta con un modo historia ambientado en un planeta extraño, donde su protagonista es un zorro llamado Tob. Tob pasará buenos momentos jugando con los Monster Numbers, monstruos que habitan el planeta, pero también encontrará otros monstruos no tan amigables que deberá tratar de evitar. El objetivo de este modo es encontrar las piezas para reparar la nave de Tob y poder volver a casa, para ello tendrá que superar distintos retos. Por otro lado, cuenta con un modo

<sup>11</sup> <https://monsternumbers.net/>

rápido el cual ofrece distintos minijuegos cada uno de ellos correspondiente a un tipo de operación.

Al empezar a jugar, lo primero que se pide es introducir nuestra edad. En función de la edad el juego adaptará las actividades propuestas. Por ejemplo, entre 4 y 5 años se jugará con los números, se aprenderá a contar, se elaborarán secuencias y se harán grupos. Si se es un poco más mayor, es decir, de 6 a 7 años se harán secuencias, grupos, sumas y restas con los números, y si aún se es más mayor, por ejemplo, entre 8 y 14 años, el juego propondrá suma, restas, multiplicaciones y divisiones. Los distintos niveles correspondientes a los ejercicios unas veces se presentan como desafíos matemáticos donde el jugador deberá hacer uso de las matemáticas para resolverlos (figura 3). Otras, simplemente como un juego donde el protagonista corre, huyendo de los enemigos, evitando obstáculos y atrapando monedas que se encontrará en su recorrido hasta alcanzar un lugar seguro (figura 3). Este tipo de nivel no incluye ningún contenido matemático, ni ninguna enseñanza es un simple modo de entretenimiento intercalado entre ejercicios matemáticos.

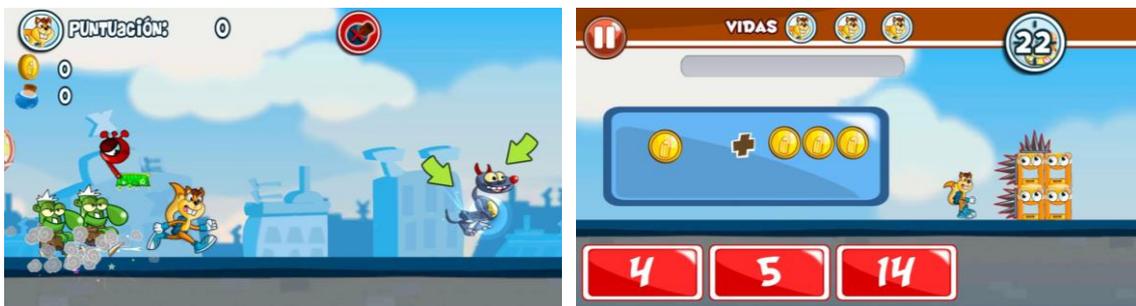


Figura 3: captura de dos escenarios distintos de Monster Numbers

La jugabilidad por lo que respecta a los niveles matemáticos es de elección, es decir, se plantea un problema, por ejemplo, se muestran en pantalla una cantidad de monedas y se ofrecen tres opciones a elegir, el jugador debe sumar las monedas y elegir la respuesta correcta. Si falla o se agota el tiempo pierde una vida de las tres que dispone, si acierta en cambio avanza en el nivel y se le presenta otro ejercicio del mismo estilo hasta que supere dicho nivel. Los ejercicios a veces se representan con monedas que el jugador debe sumar o restar, o a veces se presenta la operación directamente (del tipo  $4 + 5$ ). A medida que se avanza en el juego y se van resolviendo operaciones se obtienen insignias como recompensa por haber logrado un cierto número de aciertos o no haber cometido ningún fallo, entre otros.

Lo primero a destacar es su atractivo visual de cara a un público infantil, gráficos bien cuidados y acorde a la temática. Por lo que respecta a los contenidos principalmente se tratan sumas, restas, multiplicaciones, divisiones y algunas secuencias a completar, todos ellos con un enfoque educativo y aliciente, sin embargo, son pocos los contenidos tratados. Un punto en contra sería la dificultad y la no correcta adecuación para la edad propuesta en la descripción del juego, puesto que el juego, según sus creadores está diseñado para todas las edades, desde 4 hasta 99 años, calificando el juego como un auténtico reto para personas de entre 16 y 99 años. Esto tras probar el juego e introducir diferentes edades para comprobar las diferencias, se observa que no es así, la dificultad y contenidos no se adecuan para una edad mayor de 16 años. Esto se corrobora con los

comentarios de los jugadores, que sitúan el juego en una edad alrededor de los 8 años. Pese a no ser adecuado para edades mayores, el juego sí que resulta educativo y entretenido para niños pequeños. Otro punto a tener en cuenta en el desarrollo de un juego para móvil es el escalado en pantallas y como se adapta a las diferentes resoluciones. El juego ha sido probado en un dispositivo con *notch* y una resolución de 2246 por 1080 píxeles. Se han observado menús que no cubrían toda la pantalla lo cual hacía que se vieran otros menús por debajo y botones que se salían de los márgenes del dispositivo.

### **Mundoprimeria<sup>12</sup>:**

Es un sitio web para niños de primaria, con una gran cantidad de contenidos como cuentos, fábulas, dibujos, lecturas, juegos, entre otros para enseñar y entretener. Dentro de sus apartados presenta el de juegos de matemáticas, el cual incluye juegos de números y operaciones, magnitudes y medidas, geometría y resolución de problemas, es decir, presenta juegos para todos los bloques de contenidos que se tratan en la primaria. Dentro de cada bloque se dividen los ejercicios por cursos desde primero hasta sexto y dentro de cada curso hay diferentes ejercicios correspondientes con los contenidos que se tratan en cada uno. Además de los juegos, cada bloque tiene explicaciones detalladas con ejemplos de los contenidos tratados en los juegos.

Cada uno de los juegos es independiente del resto, tanto en historia (no tiene), como en temática (cada juego tiene unos diseños distintos), como en jugabilidad. Existen juegos donde el jugador debe arrastrar objetos, por ejemplo, descomponer un número en unidades, decenas, décimas y centésimas donde el jugador debe arrastrar cada número a su posición correcta, otros donde debe escribir el resultado de la operación que se le plantea y otros donde se le proponen soluciones y el jugador debe elegir la respuesta correcta. En la figura 4 se pueden observar de izquierda a derecha los tres tipos de juegos descritos.



Figura 4: captura de tres juegos de Mundoprimeria

<sup>12</sup> <https://www.mundoprimeria.com/juegos-educativos/juegos-matematicas>

Lo primero a destacar en la gran cantidad y variedad de ejercicios que tiene, por cada bloque hay seis cursos y por cada curso entre 11 y 20 juegos distintos, es decir, alrededor de unos 400 juegos. Sin embargo, algunos juegos se repiten entre cursos con la diferencia de que aumentan la dificultad. Las explicaciones de los contenidos son un aliciente, aunque no aporten jugabilidad, aportan enseñanza y son de utilidad para aprender antes de hacer los ejercicios. Primero te explican contenidos y luego haces ejercicios, no obstante, el juego no obliga a pasar por estos apuntes si lo que el jugador realmente desea es solo jugar. Un punto negativo para destacar es la falta del algún tipo de puntuación, recompensa o logro al superar un juego. Tampoco hay vidas ni tiempo para resolver los ejercicios, esto desde el punto de vista del jugador, no le propone un objetivo o reto que afrontar, puramente la satisfacción de completar un juego y con suerte que le haya sido útil en su formación.

### Juegos matemáticos<sup>13</sup>:

Es un videojuego gratuito para plataforma móvil, hay una versión de pago la cual no incluye anuncios. Cuenta con más de diez millones de descargas para Android y una valoración por los usuarios de 4,4 sobre 5. Posiblemente sea el más sencillo de los tres juegos presentados, tanto por su diseño minimalista como su jugabilidad (figura 5).

Lo primero que se observa al iniciar el juego son los tipos de juego que ofrece en total seis, sumas, restas, multiplicaciones, divisiones, raíces y decimales. Dentro de cada tipo, existen diferentes modos de juego. El modo aprender cuyo único propósito es mostrar ejemplos de operaciones. El modo examen, sucesión de operaciones donde se presenta una operación y se ofrecen cuatro soluciones para elegir. El modo duelo que permite jugar a dos personas en el mismo dispositivo. El modo hora donde hay que resolver las operaciones antes de que termine el juego y algunos modos más de juegos. El juego no presenta ningún tipo de historia, ni ninguna relación entre juegos, es decir, son juegos independientes donde cada uno de ellos trata un tipo de problema.



Figura 5: capturas del juego Juegos matemáticos

El juego permite elegir el idioma entre los diez idiomas disponibles, además de la dificultad de los ejercicios (fácil, medio y difícil) y la cantidad de preguntas para los modos (10, 20 o 30 preguntas para completar el nivel).

<sup>13</sup> <https://play.google.com/store/apps/details?id=com.GamesForKids.Mathgames.MultiplicationTables>

Dispone de seis tipos de operaciones que podrían resultar pocas, sin embargo, dentro de cada tipo existen los distintos modos, además de poder ajustar la dificultad y cantidad de operaciones a completar, esto hace que el juego se adapte a las necesidades o capacidades del jugador. Es un juego con una gran cantidad de jugadores, y por tanto recibe actualizaciones constantemente para añadir más contenidos y solucionar fallos que pueda tener. Como punto negativo destacaría el diseño tan minimalista y simple del juego, es verdad que cuenta con muchos colores (recurso utilizado para atraer a los niños) pero bastante pobre en imágenes y animaciones, como resultado no lo hacen especialmente atractivo. Pese a su pobre diseño, el juego cumple las expectativas de enseñar, y su personalización hace que los contenidos se adecuen a la edad del jugador. Otro punto fuerte para destacar es que no requiere de conexión a Internet para poder jugar.

## Resumen

Probar y analizar distintos videojuegos ha permitido estudiar qué características y funcionalidades eran necesarias incorporar, cuáles innecesarias o prescindibles y cuáles podrían resultar interesantes implementar en Maths. A continuación, se pasa a comparar las características de estos, para ello se ha elaborado una tabla que resume las características presentes y ausentes en los tres títulos analizados y las características que incorpora Maths.

	Monster Numbers	Mundo primaria	Juegos matemáticos	Maths
<b>Historia</b>	Sí	No	No	No
<b>Instrucciones de juego</b>	Sí	Sí	No	Sí
<b>Distintos modos de juego</b>	Sí	Sí	Sí	Sí
<b>Modos desbloqueables</b>	Sí	No	No	No
<b>Sistema de logros</b>	Sí	No	No	Sí
<b>Recompensas (puntos o monedas)</b>	Sí	No	No	Sí
<b>Multijugador local</b>	No	No	Sí	No
<b>Publicidad</b>	Sí	Sí	Sí	No
<b>Requiere conexión a Internet</b>	No	Sí	No	No
<b>Plataformas</b>	PC, Android, iOS	PC	Android	PC, Android
<b>Versión completa de pago</b>	Sí	No	Sí	No
<b>Precio</b>	2,29€	Gratuito	4,09€	Gratuito



Como podemos observar, una de las cosas importantes de Maths es que incorpora la mayoría de las características positivas de los tres juegos. Presenta instrucciones, distintos modos, no hay modos desbloqueables (lo que significa que desde el primer momento de juego el jugador tiene acceso a los diez modos disponibles), incluye logros, puntuación, no tiene publicidad, no requiere acceso a Internet para jugar, es gratis y es jugable tanto en Android como PC, además en un futuro se podría exportar a iOS gracias a la portabilidad de Unity.

A modo de resumen, este sería el análisis que ha servido para llevar al cabo el desarrollo de Maths. Lo primero a destacar sería la variedad de ejercicios distintos que tenga el videojuego, a mayor cantidad, mayor posibilidad de tener entretenido al jugador a la vez que aprende. Sobre todo, en contenidos visuales, un videojuego no debe ser repetitivo a nivel visual, es decir, puede que sean modos de juego distintos, pero todos con el mismo diseño, esto hace que el jugador vea el juego como algo repetitivo y monótono. Por tanto, hay que introducir contenidos visuales distintos entre un modo de juego u otro.

Los tres juegos analizados presentan un modo de juego de elección, donde el juego propone las soluciones y jugador elige la respuesta que cree correcta. Esto aporta dinamismo y si lo que se busca es rapidez a la hora de jugar esta es una buena elección. Muchos juegos no solo educativos, diseñados para plataforma móvil esta pensados para que, si tienes cinco minutos disponibles, puedas realizar una partida en ese tiempo y puedas salir de juego sin ninguna pantalla de carga, ni guardado, ni tiempo de espera para poder jugar, esto es un punto fuerte a tener en cuenta a la hora de desarrollar un juego para móvil. Además, como el objetivo es que el videojuego sea para dispositivo móvil en orientación horizontal, mostrar un teclado en la pantalla resultaría desagradable y molesto ya que ocultaría la mayor parte del juego y posiblemente no se vería lo que el jugador estuviera escribiendo hasta que cerrará el teclado.

Por lo que respecta a la historia del juego, no se ve como un punto clave para este tipo de videojuego educativo. Si que es verdad que el añadir una historia, haría que el jugador se sumergiera en el videojuego y tuviera ganas de continuar jugando para saber cómo termina, sin embargo, creo que el desarrollo de un juego con modo historia dificultaría la posibilidad de incluir la mayor parte de contenidos del libro de matemáticas además del tiempo que supondría elaborar una historia con fundamento adecuado a la causa. Por ello, creo que es mejor añadir como aliciente al juego retos o logros que alcanzar, trofeos que conseguir como resultado de ir superando los niveles o alcanzando cierta puntuación.

### 2.1.3 Propuesta

---

Con el presente TFG se contribuye al desarrollo de un videojuego educativo jugable mediante ordenador o dispositivo móvil Android para el aprendizaje y enseñanza de matemáticas de tercero y cuarto de primaria. De esta forma se ofrece un videojuego orientado para niños de entre ocho y diez años, pero con disponibilidad para todo el público, pues cuyo fin es que sea útil para el resto de los cursos de primaria y con la posibilidad de ser ampliado en contenidos (más detalles en el punto de trabajo futuro). Videojuego cuyos contenidos han sido extraídos de libros oficiales de matemáticas

elaborados por editoriales conocidas en el mundo académico (los libros y editoriales se nombran en el siguiente punto, fundamentos teóricos).

Este videojuego no pretende sustituir a los ya existentes videojuegos sobre matemáticas, sino, incluir las características más destacadas de estos títulos, unificando contextos y estableciendo contenidos adecuados a la edad, pues la mayoría de los videojuegos matemáticos presentan un amplio rango de edades y los contenidos no se adecuan correctamente. Es por ello por lo que Maths, propone elaborar contenidos centrándose en una edad concreta y con posibilidad de incrementarse tanto en contenidos como en plataformas de acceso en un futuro.

## 2.2 Fundamentos teóricos

---

Con el objetivo de que el proyecto tuviera la supervisión pedagógica y orientación de profesionales del sector. Se estableció contacto con distintas editoriales que elaboran libros de matemáticas de primaria, entre ellas, Santillana, Anaya y Bromera. En un principio la comunicación se intentó con correos y solicitudes a través de sus páginas web. Al no obtener respuesta a las solicitudes se estableció contacto vía llamada telefónica, pero por desgracia no estaban interesados en formar parte del proyecto.

Tras ponerse en contacto con varias editoriales y recibir la negativa a colaborar, se tuvo que buscar canales alternativos para llevar a cabo el desarrollo. La alternativa a seguir y la cual ha permitido el desarrollo adecuado del proyecto ha sido utilizar libros de matemáticas de primaria de las editoriales nombradas anteriormente. De esta forma, se asegura que los contenidos docentes del libro (teoría, ejemplos y ejercicios) ya han sido avalados por expertos. Libros que la editorial ya ha elaborado y por tanto los contenidos docentes del mismo ya cumplen con los requisitos formativos de la educación primaria y los contenidos siguen una progresión justificada pedagógicamente.

En concreto los libros consultados han sido los siguientes (en la figura 6 se pueden observar las portadas de los libros):

- +Mates 3 Primària (Voramar, ISBN: 9788491315933)
- Mate+ 3 Primaria (Serie Práctica, ISBN 9788414112298)
- Matemáticas 3 Primaria (Grazalema, ISBN: 9788491321651)
- Mate+ Matemáticas para pensar 3 Primaria (ISBN: 9788468039916)
- Mates Sèrie Pràctica 4 Primària (Voramar, ISBN: 9788491316497)
- Mate+ Matemáticas para pensar 4 Primaria (ISBN: 9788414105306)



Figura 6: portada de los libros de matemáticas utilizados

Desde la página web de Santillana<sup>14</sup> se puede acceder a un amplio catálogo de libros para consultar. La mayoría no están completos (se deben comprar) pero sí que ofrecen una muestra bien amplia de todos los temas que tienen y los ejercicios que salen en cada tema, además se puede consultar tanto la versión del libro del alumno como la versión del libro del profesor la cual incluye consejos a la hora de impartir los conocimientos. Los libros nombrados son de la Comunidad Valenciana y de Andalucía.

Debido a la extensión de los libros y la gran cantidad de contenidos que abarcan, la falta de tiempo ha hecho imposible implementar todos los contenidos de este. De tal forma que se han seleccionados los contenidos y ejercicios más relevantes a implementar de cara a hacer un videojuego que enganche al público. Una vez seleccionado los contenidos, se ha realizado un estudio sobre como adaptar ejercicios de un libro de texto a un videojuego y que continúen teniendo la misma finalidad de enseñar, pero de forma entretenida.

## 2.2.1 Teoría del Aprendizaje de Piaget

Jean Piaget es un psicólogo suizo, el cual realizó una tesis sobre el desarrollo psicológico en la infancia y la teoría constructivista del desarrollo de la inteligencia. De ahí nació lo conocido actualmente como Teoría del Aprendizaje de Piaget. Para Piaget el proceso de

<sup>14</sup> <https://santillana.es/mate-mas/>

aprendizaje solamente cobra sentido frente a situaciones de cambio. Por esa razón, aprender es en parte adaptarse a esas novedades. La Teoría del Aprendizaje explica la dinámica de adaptación mediante los procesos de asimilación y acomodación. Mediante la asimilación y la acomodación se va reestructurando cognitivamente nuestro aprendizaje a lo largo del desarrollo (reestructuración cognitiva), ambas acciones interactúan mutuamente en un proceso de equilibrio. A la hora de desarrollar Maths se ha considerado interesante incluir la Teoría del Aprendizaje de Piaget pues Piaget destaca en gran medida la importancia del juego en el proceso de desarrollo, ya que la primera tarea del educador es la de generar un interés con el que poder entender y actuar con el alumno. Maths trata de captar la atención, generar dicho interés en los niños para luego a través del juego lograr la tarea del aprendizaje.

La idea de su teoría trata de aplicarse durante el proceso de enseñanza en el aula y por eso Maths trata de seguir la teoría aplicada al videojuego. El objetivo del videojuego al igual que el del libro de matemáticas no es enseñar unas reglas y operaciones al alumno para que dé soluciones exactas a determinados problemas, sino tratar de ayudarlo en el proceso de desarrollo a lograr la competencia numérica necesaria para que sea capaz de hacer uso de sus conocimientos frente a situaciones reales de la vida cotidiana. Se pretende desarrollar la flexibilidad de pensamiento en los niños y niñas de tal modo que les permita comprender las matemáticas de una manera fácil. Desarrollar el pensamiento reversible, que les permita a los niños y niñas desplazarse con soltura y seguridad por el cálculo de operaciones contrarias entre sí ( $6+4 = 10$ ;  $10 - 4 = 6$ ;  $10 - 6 = 4$ ). Esto les favorecerá a prosperar en el cálculo mental y a entender las conexiones que se establecen entre los números. Para favorecer al desarrollo del pensamiento lógico matemático es significativo plantear las operaciones en el contexto de un entorno dificultoso, en el que el alumno tenga que imaginar un problema en función de la operación planteada. De esta forma se favorece no solo a la competencia matemática, sino que se propicia que aprendan a aprender, teniendo iniciativa por formular hipótesis y resolver problemas. (15)

## 2.3 Técnicas utilizadas para la enseñanza a través de juegos

---

El uso de videojuegos en las actividades de aprendizaje y enseñanza constata que estos instrumentos tecnológicos pueden ser utilizados como herramientas pedagógicas que ayudan, entre otras cosas, a la resolución de problemas de aprendizaje, a la mejora de las habilidades motoras y cognitivas, y al fomento de la creatividad.

La ludificación más bien conocida como gamificación (adaptación del inglés gamificación) (16) es una técnica de aprendizaje cuyo fin es trasladar la mecánica de los juegos al ámbito educativo con la finalidad de conseguir mejores resultados, ya sea por mejorar alguna habilidad, absorber mejor algunos conocimientos, o bien recompensar acciones concretas, entre otros muchos objetivos. Para llevar a cabo la gamificación se utilizan una serie de técnicas mecánicas y dinámicas extrapoladas de los juegos. (17)

Ambas técnicas forman parte del denominado canal de flujo cuyo objetivo es presentar desafíos para el jugador manteniendo el equilibrio de cada elemento del desafío entre la



habilidad y la dificultad. A medida que el jugador aprende del juego aumentando su habilidad, se debe incrementar la dificultad de manera proporcional. Tener un buen canal de flujo permite mantener un balance para mantener a los jugadores anclados al videojuego, evitando que se aburran por la facilidad o se estresen por dificultad (ver figura 7). (18)

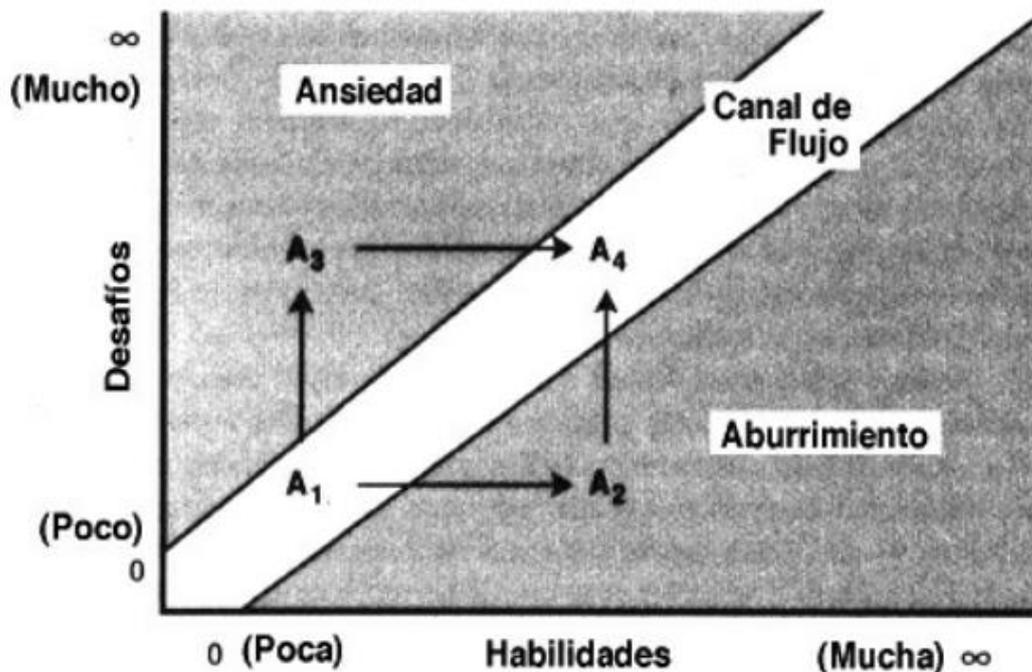


Figura 7: concepto del canal de flujo (19)

Las técnicas mecánicas son la forma de premiar al jugador en función de los objetivos alcanzados tras la superación de una sucesión de pruebas y obstáculos con las que se consigue la participación y el compromiso de los usuarios. Algunas de las técnicas mecánicas más utilizadas son las siguientes:

**Acumulación de puntos:** otorgar a acciones concretas como responder correctamente a un ejercicio o a un reto un valor habitualmente cuantitativo y acumulativo como muestra de la superación de las acciones. De este modo, se da información de progreso.

**Escalado de niveles:** establecer una sucesión de niveles que el jugador debe ir superando para ascender al siguiente. Con este sistema se premia la participación del usuario otorgándole un distintivo del resto de usuarios.

**Obtención de premios:** según se van consiguiendo distintos objetivos se van otorgando premios a modo de colección. La obtención de trofeos o insignias a modo de reconocimiento tras superar una meta sirve a los usuarios para presumir ante sus amigos u otros jugadores de esas colecciones.

**Regalos:** obsequios que se dan al jugador o jugadora de forma gratuita al superar un objetivo. La obtención de objetos de personalización que motiven al jugador a querer seguir jugando para lograrlos.

**Clasificaciones:** clasificar a los jugadores en función de objetivos o puntos conseguidos, destacando los mejores en una lista o ranking. De este modo, se explota el espíritu competitivo de los usuarios, el jugador compite por ser el mejor.

**Desafíos:** competiciones entre los jugadores, el mejor obtiene el premio o los puntos.

**Misiones o retos:** lograr resolver o superar un reto u objetivo propuesto, ya sea en equipo o solo.

Por otro lado, las técnicas dinámicas implican a la motivación del jugador por querer jugar y avanzar en el progreso del juego logrando los objetivos que se lo propongan. A continuación, se exponen algunas de las técnicas dinámicas más utilizadas:

**Recompensa:** elementos especiales del juego, se busca retar al jugador a cambio de ofrecerle una condición especial o darle un poder que normalmente no está disponible, es decir, obtener un beneficio merecido.

**Estatus:** establecerse en un nivel jerárquico social valorado, mediante torneos y ligas de videojuegos tanto multijugador en línea como fuera de línea.

**Logro:** pequeños requerimientos que pueden variar en dificultad y nutren los videojuegos, añadiendo fundamentalmente un seguimiento al progreso de los jugadores a la vez que les otorga un distintivo como marca de superación o satisfacción personal.

**Competición:** por el simple afán de competir e intentar ser mejor que los demás, dinámica muy presente en los juegos de disparos y deportes.

En función de la dinámica seguida, se deberán explorar más unas técnicas mecánicas que otras.

## 2.4 Técnicas aplicadas a Maths

---

Maths es un videojuego donde se aprende jugando, es por ello clave, tener presente el incremento de la habilidad del jugador para proporcionándole un incremento de dificultad proporcional a la habilidad de este. Cuanto más se juegue mayor será la capacidad del jugador y mayor puntuación obtendrá en cada partida. Maths presenta un desafío permanente que trata de involucrar al jugador para que no deje jugar. Además, como refuerzo al desafío otorga logros tras alcanzar cierta puntuación y cantidad de aciertos. Los logros no solo pretenden lograr la satisfacción personal y superación del jugador, sino mantenerlo sumergido para que siga jugando hasta conseguirlos todos. En segundo plano y como consecuencia positiva de jugar a Maths, el logro también sirve como marca de superación del jugador, que no solo juega por entretenimiento, sino porque le motiva superarse día a día y gracias a Maths aprende matemáticas de una forma más llevadera de lo que podrían ser unas clases teóricas.

El videojuego desarrollado no presenta distintos niveles, pero sí aumento de la dificultad de cada modo de juego a medida que se acierten las respuestas y se consiguen puntos. Este aumento de dificultad hace que el jugador perciba el juego como algo lineal, en el que la dificultad aumenta gradualmente. El juego es de un único jugador y no presenta rankings para saber la posición respecto a otros jugadores. Pero cada nivel muestra la



puntuación máxima obtenida en él, este sistema permite al jugador compararse con otros jugadores. Pongamos un ejemplo, una clase de alumnos donde el profesor ha dicho que dará una recompensa al que obtenga mayor puntuación en el juego de suma, esto hará que los niños jueguen y compitan entre ellos por obtener la máxima puntuación. En esta situación, al no haber rankings podrían ocurrir dos cosas, por una parte, que nuestros compañeros nos enseñarán su puntuación para ver si están por delante de nosotros y, en tal caso, seguir jugando para intentar sobrepasarles o, por otra parte, que estos no nos mostraran su puntuación y, por tanto, la única opción para ser el mejor sería seguir jugando para mejorar cada vez más nuestra puntuación y con esto lograríamos el objetivo del juego, aprender.

El videojuego presenta un sistema de trofeos, en el cual se pueden obtener trofeos o insignias tras alcanzar una cierta puntuación o lograr una cantidad específica de aciertos en cualquier modo de juego. En concreto hay establecidos 66 logros, por cada modo de juego hay disponibles seis logros. Para ser más exactos existen tres logros sobre puntuación, al alcanzar 100, 1.000 y 10.000 puntos respectivamente, y tres más al lograr 10, 100 y 1.000 aciertos, por cada uno de ellos se consigue un trofeo distinto. En total existen 10 modos de juegos, un total de 60 logros, los seis restantes se obtienen al superar todos los logros de puntuación de todos los modos, alcanzando 100, 1.000 o 10.000 puntos en los modos, lo mismo ocurre con los aciertos. Tanto los logros de alcanzar 10.000 puntos y 1.000 aciertos se presentan como reto personal para aquellos jugadores más avanzados que deseen un reto realmente difícil y sean capaces de completar el videojuego al 100%.

El diseño visual y auditivo también juegan un papel importante, ya que harán de reclamo para el jugador. A la hora desarrollar el videojuego se debe tener en cuenta la edad de los jugadores a los que va dirigido, esto determinara el estilo a seguir, colores, imágenes, animaciones, los diseños del videojuego en general. Se debe tener en cuenta la sensación que se pretende provocar en los jugadores. Maths va destinado a niños de alrededor de nueve años, por lo que se debe transmitir sensación de bienestar. Para ello se deben utilizar colores como el azul o el verde, para captar la atención de los jugadores pero siempre en tonalidades no demasiado estridentes que puedan causar el efecto contrario.

El verde aporta concentración junto con el azul que ayuda a mejor la comprensión lectora. Por ello el color del menú principal es un azul en distintos tonos con frutas de colores que aportan vitalidad y energía. Las frutas, son el principal recurso visual utilizado para captar la atención del jugador, por ello, en cada juego se propone un objetivo relacionado con ayudar a los fruteros y fruterías con la fruta, a contarla, transportarla o cargarla, entre otros objetivos. También se ha utilizado un turquesa como color de fondo para todos los modos de juegos. Se ha jugado bastante con las tonalidades de azul y verde, para que influyan de forma correcta en el proceso de aprendizaje, pero que a la vez no sea contraproducente ni repetitivo visualmente para el jugador. Se ha evitado utilizar el rojo como color principal de ningún menú ni apartado del videojuego, no es un color adecuado para zonas de estudio, ya que este color tiende a estimular y sobreexcitar, por lo que no es aconsejable para personas hiperactivas. Con el naranja ocurre algo similar, en tonalidades más claras es un color acogedor que hace que la atención del alumno aumente, además de ayudar a promover la comodidad e incluso reforzar la mejor del funcionamiento neuronal. Sin embargo, una tonalidad de naranja intensa puede provocar efectos adversos como el rojo. (20)

## 2.5 Contenidos a implementar

---

El videojuego se divide en diferentes modos de juegos correspondientes a los distintos temas de libro de matemáticas, de esta forma se obtiene un videojuego versátil, que dota al jugador de una mayor libertad. Así pues, el jugador tiene la opción de ir directamente al modo de juego cuyos contenidos quiera aprender o estudiar o lo entretengan más. A continuación, se pasa a detallar cada modo de juego, los contenidos que abarca y las competencias que cubre.

Tras consultar diferentes libros de matemáticas de tercero y cuarto de primaria para llevar a cabo el desarrollo del videojuego, se observa la tendencia de estos libros a estructurarse en cinco bloques de contenidos. Un primer bloque de numeración, a continuación, un bloque de cálculo mental y operaciones, otro de resolución de problemas, medida y geometría y por último un bloque de tratamiento de la información. Dicha división en cinco bloques sufre variaciones en libros de editoriales distintas, pero llegan a abordar los mismos contenidos.

### **Estructura compartida por los 10 juegos**

Todos los juegos han sido creados siguiendo la misma estructura de diseño. Al seleccionar un juego, lo primero que al jugador se le muestra es el objetivo del juego (qué debe hacer para ganar), y un mensaje de recuerda. Con cada objetivo, se le plantea al jugador a una situación problemática (Piaget) la cual requiere de su ayuda para ser solucionada. El mensaje de recuerda es aleatorio y se corresponde con el juego seleccionado, es decir, cada juego tiene una cantidad de mensajes distintos que sirven de ayuda al jugador tanto como para repasar contenidos del temario, recordar propiedades matemáticas, aprender algo nuevo, o simplemente un consejo a la hora de jugar.

Una vez iniciada una partida, el jugador dispone de un tiempo específico (tiempo que varía en función del juego y su dificultad) para responder antes de que se le termine y pierda la partida. Al acertar una respuesta este tiempo gana un incremento que le permite seguir jugando, si en cambio, falla la respuesta, no recibe una penalización en tiempo sino en puntos y vidas. Inicialmente se dispone de tres vidas, a medida que el jugador falle sus respuestas irá perdiéndolas hasta llegar a cero y terminarse la partida. No existe ninguna forma de recuperar vidas, así que el jugador debe tomar su tiempo para pensar la respuesta, pero teniendo en cuenta que no se le agote. La puntuación es la forma de premiar al jugador, para ello, se ha implementado un sistema de puntos variable en función del tiempo. Este sistema es percibido por el jugador a través de una pirámide de fruta que a medida que avanza el tiempo disminuye, a menor cantidad de fruta menor será la puntuación, pero siempre se otorgan puntos positivos por cada acierto, nunca se penaliza de forma negativa un acierto. Con cada respuesta la pirámide vuelve a empezar junto con el nuevo enunciado. La única forma de perder puntos es al fallar la respuesta.

Durante la partida el jugador puede pausar la partida, en este menú puede consultar el objetivo del juego por si no lo recuerda o no entiende que debe hacer, además tiene la opción reiniciar la partida o salir de ella si lo desea. Por último, al finalizar una partida tras haber perdido las tres vidas o haberse agotado el tiempo, se muestran los resultados



finales. El jugador puede ver la puntuación obtenida, los aciertos y el tiempo total que ha durado, también dispone de la opción de volver a iniciar una partida del mismo tipo o ir al menú de contenidos para elegir un juego distinto.

## 2.3.1 Numeración

---

El videojuego se ha centrado más en el bloque numeración, es el tema que dispone de más variedad de ejercicios (diferentes juegos) ya que este bloque es la base del aprendizaje para poder avanzar en los otros bloques de contenidos. En el videojuego se han implementado seis juegos sobre numeración cuyos contenidos pasan a detallarse a continuación:

1. El número mayor y el número menor empleando los signos  $>$ ,  $<$ ,  $=$ .
2. Los números de tres cifras en los que se tiene que indicar las unidades, decenas y centenas hasta seis cifras donde se debe indicar las unidades, decenas y centenas de millar.
3. Los números impares y pares.
4. Los números anteriores y posteriores.
5. Los números romanos.
6. Las fracciones, que contienen a su vez la lectura, representación e interpretación de fracciones.

El bloque de numeración trata de cubrir una serie de competencias que trataría cualquier libro docente de matemáticas elaborado por una editorial. A través de este bloque se persigue lograr la alfabetización numérica y operacional, tratando de obtener las capacidades necesarias para afrontarse satisfactoriamente a situaciones de la vida cotidiana en las que participen los números, sus relaciones y las operaciones, comprendiendo de un modo correcto el concepto de los números, su valor y sus usos, de modo que pueden ser interpretados en contextos numéricos de la vida diaria. Así pues, una correcta interpretación de la numeración permite la obtención de información clara y entendible, directamente o mediante la comparación, la estimación y el cálculo mental o escrito.

### **Juego mayor o menor**

El objetivo del juego es indicarle al frutero que cajas debe cargar indicándole en cuál de ellas hay más fruta. Para llevar a cabo la tarea, se muestra como enunciado dos números y el jugador dispone de tres opciones a elegir, mayor que (se muestra con el signo  $>$ ), menor que (con el signo  $<$ ) o igual que (con el signo  $=$ ). Entre ambos números hay un espacio para ubicar el signo correcto, el objetivo del jugador es elegir cuál de los tres símbolos es la respuesta correcta.

Los números que inicialmente se muestran están formados por tres cifras, a medida que el jugador acierte la dificultad aumenta. Al llegar a 20 aciertos, los números pasan a tener 4 cifras, cuando se logren 40 aciertos pasan a tener 5 cifras, a los 60 aciertos vuelven a ser 3 cifras, pero esta vez con un decimal y al llegar a 80 aciertos se alcanza la dificultad

máxima con 4 cifras y un decimal. De esta forma se consigue recorrer los tipos de comparación de valores que se realizan en los cursos de tercero y cuarto de primaria.

### **Juego fracciones**

En este juego se debe ayudar al frutero indicándole la porción de fruta que debe entregar a sus clientes. Para ello, se muestra como enunciado una porción de una fruta, y se dispone de tres opciones a elegir, cada opción es una fracción distinta. El objetivo del jugador es saber que porción de fruta hay representada y elegir la fracción correcta.

La porción de fruta que se muestra viene delimitada por el numerador y denominador de la fracción. El valor para el numerador oscila entre uno y nueve, mientras que el valor para el denominador oscila entre uno y diez. Se han establecido estos márgenes para el numerador y denominador pues al representar la fracción de fruta con una imagen, poner valores de dos cifras dificultaría el juego tanto al crear las imágenes de las fracciones como al jugador identificar la fracción representada. Cada fracción está representada por cuatro frutas distintas, de esta forma se consigue un amplio surtido de fracciones para conseguir diversidad y no hacer monótono el juego. Para ajustar la dificultad, inicialmente se muestra la fruta entera delimitada por rayas para indicar en cuantos trozos está dividida (esto indica el denominador de la fracción), la parte coloreada (indica el numerador de la fracción) representa la fracción que se pide al jugador como respuesta. Al llegar a 40 aciertos estas rayas delimitadoras desaparecen aumentando la dificultad de juego ya que el jugador tiene que pensar cual es el denominador y numerador simplemente viendo la porción de fruta coloreada.

### **Juego par o impar**

Para este juego el objetivo es cargar la camioneta del frutero indicándole si debe de cargar un número para o impar de frutas. Para ello, se muestra como enunciado una cantidad de frutas que oscila entre 1 y 30, y se ofrecen dos posibles soluciones, par o impar. El objetivo del jugador es contar las frutas que se muestran y decidir si es un valor par o impar.

Para ajustar la dificultad, se juega con la cantidad de frutas que se muestran, ya que a mayor cantidad mayor dificultad. La cantidad mostrada inicialmente oscila entre 1 y 5 frutas, cada cinco aciertos este margen aumenta hasta llegar a 30 frutas, margen que se alcanza a llegar a los 60 aciertos. Se ha establecido un margen de 30 frutas y no más porque a partir de 20 frutas en la pantalla y con posiciones aleatorias el tiempo empleado para hacer el recuento y determinar la solución suele ser mayor que el tiempo ganado al acertar. Al alcanzar el intervalo entre 1 y 30 el jugador debe aprovechar cuando haya poca fruta para así ganar tiempo, tiempo que le será útil cuando le salga una cantidad mayor de frutas y deba emplear el tiempo ganado.

### **Juego anterior o posterior**

En este juego el objetivo es indicar la tienda a la que debe de llevar la fruta la frutera. Para llevar a cabo la tarea, se muestra como enunciado un valor numérico junto con un cartel que indica el resultado esperado, es decir, si se quiere el valor posterior o anterior al valor mostrado. El objetivo del jugador es entender que valor se le está pidiendo y decidir cuál es la opción correcta.



El número mostrado oscila entre los valores 1 y 1.000 (rango de valores en los que oscilan ejercicios de este estilo en los libros de matemáticas consultados). Para ajustar la dificultad, dos de las soluciones que se proponen siempre varían en más o menos uno del valor del enunciado, la tercera opción propuesta varía en más o menos diez. De esta forma se consigue ajustar más las opciones, forzando al jugador a pensar la respuesta.

### **Juego unidades, decenas y centenas**

El juego tiene como objetivo ayudar al frutero a localizar las naranjas para cargarlas en la camioneta. Así pues, se muestra inicialmente como enunciado tres frutas cada una con un número, cuyo valor no se utiliza para resolver el ejercicio, lo importante es la posición que ocupa. El juego dispone de tres opciones a elegir, unidades, decenas y centenas. Las frutas que se muestran pueden repetirse excepto una de ellas que siempre es una naranja y nunca puede haber más de una. El objetivo del juego es elegir en que posición está la naranja, si en las unidades, en las decenas o en las centenas. Con este juego se logra aprender el valor posicional que toman los dígitos dentro de un número.

Inicialmente se muestran tres frutas, a medida que el jugador acierte la dificultad aumenta. Cada 20 aciertos, se añade una fruta hasta llegar a un máximo de seis frutas donde puede variar la posición de la naranja. Las soluciones también se incrementan y surgen tres opciones más, las unidades de millar, las decenas de millar y las centenas de millar. Asimismo, las opciones de respuesta ofrecidas desde el inicio de la partida varían su posición, es decir, no se muestran en orden, ya que sería muy fácil si mantuvieran el orden real de centena, decenas y unidades. Hasta tercero se alcanzan las decenas de millar y en cuarto se alcanzan las centenas de millar (dificultad máxima del juego).

### **Juego números romanos**

El objetivo de este juego es traducir el número romano a arábigo para saber la cantidad de fruta a la que hace referencia. Para ello, se muestra como enunciado un número romano, y se ofrecen tres posibles soluciones, cada una de ellas con un número arábigo. El objetivo del jugador es saber qué número arábigo representa el número romano y elegir la respuesta idónea.

La dificultad viene establecida por el rango de valores sobre los que oscila el número romano mostrado, inicialmente este rango puede ser un valor entre 1 y 1000. Al alcanzar los 20 aciertos, el rango llega hasta 3999, este valor es el máximo que se puede alcanzar con los siete símbolos que ofrece la numeración romana (mayor número romano representado en el juego, MMMCMXCIX). Aunque se puedan expresar cantidades mayores poniendo una raya sobre los símbolos, esto ya queda fuera del alcance del curso.

## **2.3.2 Cálculo mental y operaciones**

---

Posiblemente el bloque más apreciado a nivel social por su influencia en nuestra vida cotidiana, de ahí que las familias tengan un especial interés en que sus hijos aprendan a sumar, restar, multiplicar y dividir correctamente tan pronto como sea posible. En el videojuego se han implementado cuatro juegos sobre cálculo mental y operaciones cuyos contenidos pasas a describirse a continuación:

1. Sumas hasta 100 y 1000, con tres cifras, los términos de la suma, la propiedad conmutativa y asociativa.
2. Restas hasta 100 y 1000, con tres cifras, los términos y prueba de la resta.
3. La multiplicación hasta 100, 1000 y 10000, los términos, la propiedad conmutativa y asociativa y la comprobación.
4. La división hasta 100 y 1000, exacta y entera, sus términos y la comprobación.

El bloque de cálculo mental y operaciones trata de cubrir en primer lugar, la comunicación lingüística, incorporando al lenguaje habitual la terminología de la suma, resta, multiplicación, división y de sus propiedades. Para llevar a cabo esto, el juego siempre utiliza el lenguaje específico de la materia para referirse a ella. Otra competencia tratada es la matemática, donde se abordan las propiedades de los números. Algunas propiedades tratadas son las siguientes: la propiedad conmutativa y asociativa de la suma o la multiplicación para resolver cálculos y situaciones problemáticas, utilizar de forma correcta los algoritmos de las operaciones básicas, saber y aplicar la prioridad de la multiplicación sobre la suma o la resta en operaciones combinadas, aplicar la jerarquía de las operaciones y los paréntesis a la resolución de cálculos, etc. Algunas de estas propiedades se abordan directamente sobre el juego, es decir, el jugador debe hacer uso de ellas para acertar las respuestas, en cambio otras se explican en los mensajes de recuerdo que se muestran al inicio de las partidas.

### **Juego suma**

El juego de suma tiene como objetivo salvar a las personas antes de que un dragón se las coma. Para ello, el jugador debe realizar una suma que le proporcionará la cantidad de fruta exacta para alimentar al dragón y así salvar a las personas. Para este juego se muestra como enunciado dos valores que se deben sumar y cuyo valor sumado oscila entre 10 y 1000 (valores que toman los ejercicios de sumas en los libros de matemáticas consultados). El objetivo del jugador es realizar la suma e ir colocando el resultado en una especie de candado de tres ruedas donde cada una de ellas se corresponde a las unidades, decenas y centenas. Una vez hecha la suma y colocado los valores, debe darle a un botón para comprobar la respuesta y pasar a la siguiente suma.

### **Juego resta**

El juego de resta tiene el mismo objetivo que el juego de suma, pero con la diferencia de que en este caso debe realizar una resta para calcular la cantidad de fruta exacta. Para este juego se muestra como enunciado dos valores que se deben restar y cuyo valor oscila entre 10 y 1000 (valores que toman los ejercicios de restas en los libros de matemáticas consultados). El objetivo del jugador es realizar la resta e ir colocando el resultado en una especie de candado de tres ruedas donde cada una de ellas se corresponde a las unidades, decenas y centenas. Una vez haya hecha la resta y colocado los valores debe darle a un botón para comprobar la respuesta y pasar a la siguiente resta.

### **Juego multiplicación**

El objetivo de este juego es ayudar al frutero a realizar el recuento de toda la fruta que tiene en las cajas. Para ello, se muestra como enunciado dos valores correspondientes al multiplicando y el multiplicador (como ayuda visual también se muestran un número de cajas equivalentes al valor del multiplicador) de una multiplicación. El objetivo del jugador es realizar la multiplicación y colocar la solución en el mismo mecanismo



utilizado en los juegos de suma y resta, con la diferencia de que esta vez se añade una rueda más para las unidades de millar.

La dificultad viene definida por el valor del multiplicando (inicialmente oscila entre 1 y 300) y el valor del multiplicador (inicialmente oscila entre 1 y 4). Cada 10 aciertos se incrementan ambos valores, hasta alcanzar un rango máximo entre 1 y 999 para el multiplicando y un rango entre 1 y 10 para el multiplicador tras alcanzar 40 aciertos. Los valores establecidos tanto para el multiplicando como para el multiplicador han sido extraídos de ejercicios similares de los libros. Tras probar el videojuego con niños se decidió ajustar la dificultad de este juego en la medida en la que se incrementan los valores del multiplicando y multiplicador, para que se incrementen más lentamente.

### **Juego división**

El objetivo de este juego es ayudar al frutero a repartir la fruta entre todas las cajas disponibles indicándole la cantidad de fruta exacta que debe de poner en cada caja. Para ello, se muestra como enunciado dos valores correspondientes al dividendo y el divisor (como ayuda visual también se muestran un número de cajas equivalentes al valor del multiplicador) de una división. El objetivo del jugador es realizar la división y colocar la solución en el mismo mecanismo utilizado en los juegos de suma y resta.

La dificultad viene definida por el valor del dividendo (inicialmente oscila entre 1 y 300) y el valor del divisor (inicialmente oscila entre 1 y 4). Cada 10 aciertos se incrementan ambos valores, hasta alcanzar un rango máximo entre 1 y 999 para el dividendo y un rango entre 1 y 10 para el divisor tras alcanzar 40 aciertos. Los valores establecidos tanto para el dividendo como para el divisor han sido extraídos de ejercicios similares de los libros. Tras probar el videojuego con niños se decidió ajustar la dificultad de este juego en la medida en la que se incrementan los valores del dividendo y divisor, para que se incrementen más lentamente.

## 3. Análisis

---

Dentro del proceso de construcción de software, las primeras actividades consisten en elicitar, analizar, definir, especificar y validar los requisitos del software. A este proceso se le denomina Ingeniería de Requisitos (IDR) y ha sido el proceso seguido para llevar a cabo el desarrollo del proyecto.

El proceso de IDR es un conjunto estructurado de actividades para el desarrollo de los requisitos del sistema. Sin embargo, el proceso se debe ajustar a las necesidades de cada proyecto y según la complejidad de este, las actividades pueden organizarse en fases.

### 3.1 Elicitación de requisitos

---

La primera etapa de la IDR es la elicitación, también conocida como identificación de los requisitos. Esta fase, presenta una serie de actividades para entender los objetivos y motivos para el desarrollo del sistema software propuesto. Además, presenta actividades para identificar los requisitos que el sistema software resultante debe satisfacer para alcanzar los objetivos.

Se han seguido una serie de técnicas de elicitación tradicionales. Debido a la pandemia mundial ha sido inviable conseguir acceso a una clase de alumnos de tercero o cuarto de primaria. Por esta razón, aplicar las técnicas tradicionales como son entrevistas, mapas de empatía o técnicas de grupo con la clase de alumnos y los profesores no ha sido una opción. Además, realizar este tipo de técnicas con una o dos clases de un centro educativo tampoco habría sido una muestra suficientemente grande como para representar a la mayoría. Por este motivo, al no tener acceso a los *stakeholders*, se ha optado por otras técnicas de elicitación tradicionales como son la introspección y el análisis de documentos, además de técnicas basadas en la representación como son los casos de uso. Además, se han utilizado técnicas como el modelo de dominio para establecer un vocabulario común del sistema a desarrollar y el diagrama de contexto para determinar los actores, los límites y el alcance del sistema.

Para llevar a cabo la técnica de introspección junto con el análisis de documentos, se ha estudiado el dominio del problema, el estado actual y se ha realizado un estudio cuyos resultados puede apreciarse en el punto 2 de esta memoria. Tras realizar el estudio, se han definido las características del sistema a desarrollar. La desventaja de la introspección es que al no tener acceso a los *stakeholders* puede resultar difícil reflejar las necesidades de los usuarios, por este motivo, se ha combinado con otras técnicas con el objetivo de definir los requisitos correctos y poder desarrollar el videojuego educativo adecuado al contexto educativo.

### 3.1.1 Glosario

---

A continuación, se pasan a definir los términos utilizados en el modelo de dominio (ver figura 8) y que son de vital importancia para la comprensión del análisis que se realiza en las fases de desarrollo.

**Jugador:** representa a la persona o usuario que interactúa con el sistema.

**Videojuego:** representa el sistema con el cual el jugador interactúa.

**Juego:** representa cada una de las partidas que se pueden desarrollar dentro del videojuego.

**Vida:** representa el número de fallos que se pueden realizar antes de perder un juego.

**Tiempo:** representa el tiempo de duración del juego.

**Punto:** representa el premio conseguido al conseguir un acierto en el juego.

**Acierto:** representa el número de veces que el jugador responde correctamente en el juego.

**Escenario:** representa el ambiente donde se desenvolverá el juego.

**Trofeo:** representa el logro conseguido tras alcanzar una determinada puntuación o cantidad de aciertos en el juego.

**Opciones:** representa las opciones que ofrece el sistema, como configurar el audio y pausar el juego.

### 3.1.2 Modelo de dominio

El modelo de dominio representa los términos y relaciones de interés del dominio del problema. Hace referencia al sistema concreto y particular que se desea desarrollar. Un modelo de dominio utiliza un diagrama de clases UML sin servicios y solo con los atributos más relevantes.

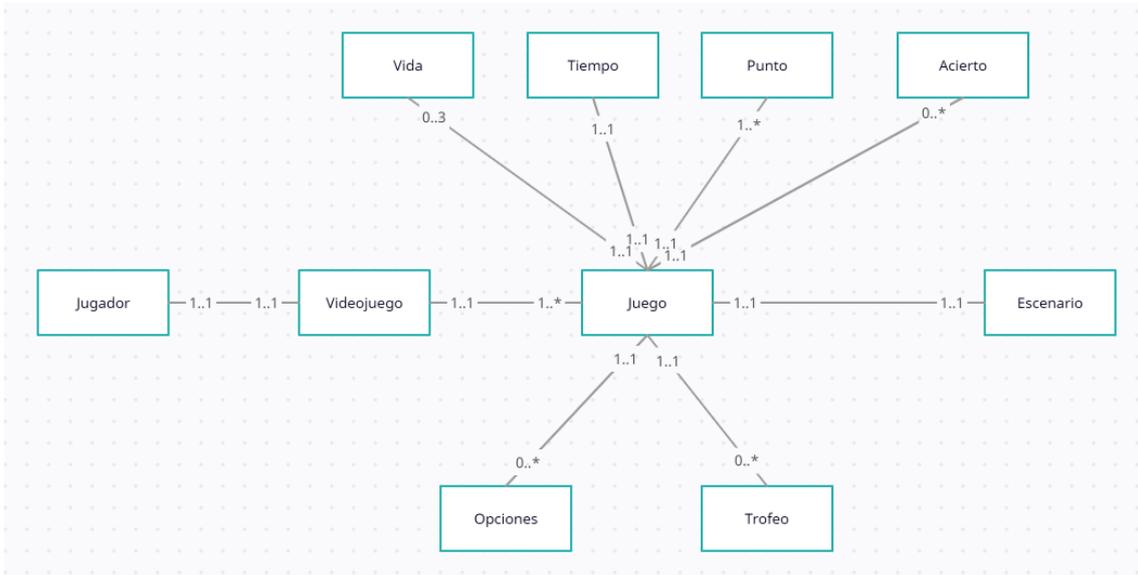


Figura 8: diagrama de dominio

### 3.1.3 Diagrama de contexto

El diagrama de contexto define los límites del sistema y los actores que interactúan con el sistema. En el sistema el único actor que puede interactuar es el jugador (figura 9).



Figura 9: diagrama de contexto

## 3.2 Identificación de requisitos

---

Una vez realizado el análisis se pasa a definir los requisitos, tanto lo funcionales como los no funcionales y posteriormente se utilizarán los casos de uso para hacer una especificación semiformal de los requisitos funcionales

Los requisitos se dividen en dos tipos de requisitos:

- **Requisitos funcionales:** describen las funciones que lleva a cabo el software, cómo debe reaccionar ante ciertas entradas y cómo debe comportarse en situaciones particulares. En otras palabras, los requisitos funcionales son las capacidades del sistema. Los requisitos funcionales de un sistema coinciden con el modelo de casos de uso.
- **Requisitos no funcionales:** describen las restricciones sobre las funciones o servicios ofrecidos por el sistema. A su vez, también definen cualidades o restricciones para el sistema, para el proceso de desarrollo o para el uso, entre otros.

### **Requisitos funcionales**

El videojuego a desarrollar debe permitir a los jugadores gestionar la forma en la que quieren aprender a la vez que se divierten. Debe mostrar un menú donde el jugador pueda elegir contenidos para jugar, configurar el sonido, consultar los trofeos o abandonar el juego. Debe permitir elegir los contenidos sobre lo que se quiere jugar, en función de los contenidos elegidos, el videojuego mostrará unos modos de juego u otros. Una vez elegidos los contenidos se deben visualizar los modos de juego, de tal forma que el jugador pueda elegir. Una vez seleccionado el modo de juego se debe iniciar la partida mostrando las instrucciones al jugador. Dentro de la partida se debe poder pausar y silenciar en cualquier momento. Desde el menú de pausa se debe poder reiniciar, abandonar o reanudar la partida. Al finalizar una partida se debe mostrar los resultados de la misma, permitir al jugador volver a empezar otra partida igual o permitirle volver al menú de selección de modo para seleccionar el próximo modo al que quiera jugar. Debe permitir consultar los trofeos, mostrando los que el jugador ha conseguido y para los que aún no ha conseguido, se debe mostrar el requisito para lograrlo. Debe permitir ajustar tanto el volumen de la música como el volumen de los efectos del juego.

### **Requisitos no funcionales**

El videojuego deberá ser compatible con diferentes teléfonos móviles con sistema operativo Android o para ordenador con sistema operativo Windows. Deberá ser compatible con diferentes resoluciones de pantalla, adaptándose según las necesidades de forma correcta. Los textos deberán ser legibles, ya sea por tamaño, color de la letra o del fondo. Los botones en los que se utilicen iconos, estos deberán dar un significado claro de la funcionalidad del botón, no deberían llevar a confusión. La interfaz deberá ser sencilla e intuitiva, para que usuarios con bajos conocimientos tecnológicos puedan usarla sin que les suponga un esfuerzo adicional.

### 3.3 Especificación casos de uso

A continuación, se muestran los casos de uso para el jugador (figura 10), cuyas principales funcionalidades serán: iniciar partida, pausar partida, reanudar partida, reiniciar partida, abandonar partida, salir del juego, consultar trofeos, silenciar juego, configurar efectos y música de fondo, comprobar respuesta (cuando el jugador introduce una solución).

#### Diagrama de casos de uso

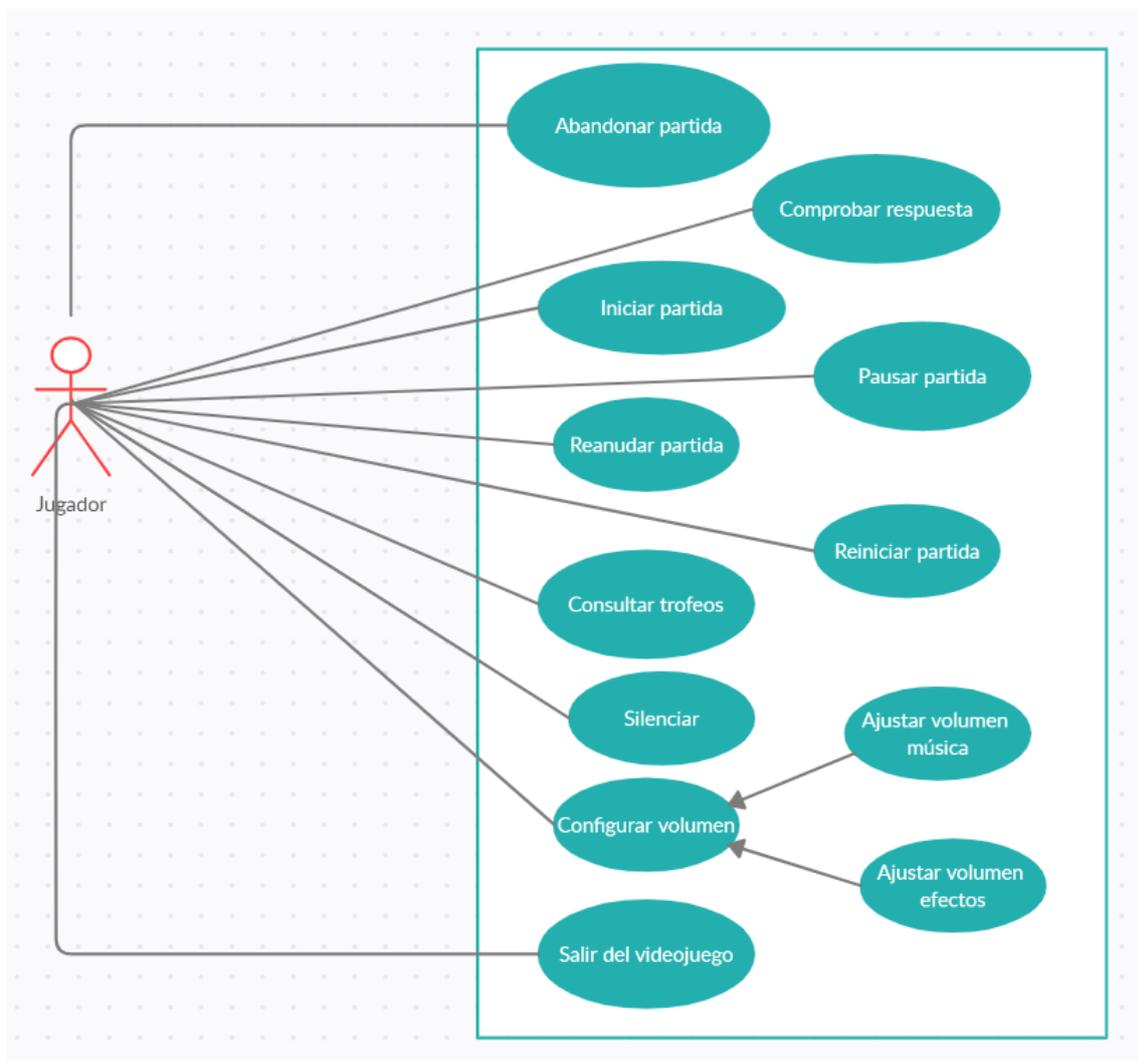


Figura 10: diagrama de casos de usos

## Casos de uso

<b>J01: Iniciar partida</b>	
<b>Descripción</b>	Empezar a jugar una partida
<b>Actor</b>	Jugador
<b>Precondición</b>	Haber accedido a los distintos modos de juego
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador selecciona un modo de juego.</li> <li>2. El sistema muestra las instrucciones del juego.</li> <li>3. El jugador pulsa la opción de continuar.</li> <li>4. El sistema muestra el juego.</li> </ol>
<b>Relaciones</b>	

<b>J02: Pausar partida</b>	
<b>Descripción</b>	Detener la ejecución de la partida
<b>Actor</b>	Jugador
<b>Precondición</b>	Haber iniciado una partida
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador pulsa el botón de pausa.</li> <li>2. El sistema oculta la partida en juego y detiene el tiempo (según el modo de juego) y muestra el menú de pausa.</li> </ol>
<b>Relaciones</b>	Requiere a J01

<b>J03: Reanudar partida</b>	
<b>Descripción</b>	Continuar jugando la partida
<b>Actor</b>	Jugador
<b>Precondición</b>	Haber pausado la partida
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador pulsa el botón de continuar.</li> <li>2. El sistema oculta el menú de pausa y muestra la partida en juego reanudando el tiempo (según el modo de juego).</li> </ol>
<b>Relaciones</b>	Requiere a J02

<b>J04: Reiniciar partida</b>	
<b>Descripción</b>	Volver a empezar de cero la misma partida que se estaba jugando
<b>Actor</b>	Jugador
<b>Precondición</b>	Haber pausado la partida o estar en el menú de final de partida
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador pulsa el botón de reiniciar partida.</li> <li>2. El sistema vuelve a iniciar la partida (caso de uso J01 iniciar partida).</li> </ol>
<b>Relaciones</b>	Requiere a J02

<b>Jo5: Abandonar partida</b>	
<b>Descripción</b>	Salir de una partida y volver al menú de modos de juego
<b>Actor</b>	Jugador
<b>Precondición</b>	Haber pausado la partida
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador pulsa el botón de abandonar partida.</li> <li>2. El sistema muestra un mensaje de confirmación para salir de la partida.</li> <li>3. El jugador confirma la solicitud.</li> <li>4. El sistema sale de la partida y muestra el menú de modos de juego</li> </ol>
<b>Relaciones</b>	Requiere a Jo2

<b>Jo6: Salir del juego</b>	
<b>Descripción</b>	Salir del videojuego cerrando la aplicación
<b>Actor</b>	Jugador
<b>Precondición</b>	Haber accedido al videojuego
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador pulsa el botón de salir del juego.</li> <li>2. El sistema muestra un mensaje de confirmación para salir del juego.</li> <li>3. El jugador confirma la solicitud.</li> <li>4. El sistema cierra el juego,</li> </ol>
<b>Relaciones</b>	

<b>Jo7: Consultar trofeos</b>	
<b>Descripción</b>	Ver los logros que el jugador ha conseguido
<b>Actor</b>	Jugador
<b>Precondición</b>	Estar en el menú principal del juego
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador accede al menú de trofeos.</li> <li>2. El sistema muestra los trofeos conseguidos y los no conseguidos muestra los requisitos para obtenerlos.</li> </ol>
<b>Relaciones</b>	

<b>Jo8: Silenciar</b>	
<b>Descripción</b>	Detener cualquier sonido del juego
<b>Actor</b>	Jugador
<b>Precondición</b>	Los sonidos están activos
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. En cualquier pantalla del juego el jugador pulsa el botón de silenciar.</li> <li>2. El sistema cancela cualquier sonido.</li> </ol>
<b>Relaciones</b>	

<b>J09: Ajustar música de fondo</b>	
<b>Descripción</b>	Ajustar el volumen de la música de fondo del juego
<b>Actor</b>	Jugador
<b>Precondición</b>	Estar en el menú principal
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador accede al menú de ajustes.</li> <li>2. El sistema muestra una barra indicando el volumen actual de la música.</li> <li>3. El jugador mueve la barra horizontalmente para establecer un mayor o menor volumen de la música de fondo.</li> </ol>
<b>Relaciones</b>	

<b>J10: Ajustar efectos</b>	
<b>Descripción</b>	Ajustar el volumen de los efectos del juego
<b>Actor</b>	Jugador
<b>Precondición</b>	Estar en el menú principal
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador accede al menú de ajustes.</li> <li>2. El sistema muestra una barra indicando el volumen actual de los efectos.</li> <li>3. El jugador mueve la barra horizontalmente para establecer un mayor o menor volumen de los efectos del juego.</li> </ol>
<b>Relaciones</b>	

<b>J11: Comprobar respuesta</b>	
<b>Descripción</b>	Verificar la respuesta introducida por el jugador
<b>Actor</b>	Jugador
<b>Precondición</b>	Estar jugando una partida de Suma, Resta, Multiplicación o División
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. El jugador realiza la operación e introduce el valor. Seguidamente pulsa el botón de comprobar.</li> <li>2. El sistema indica al jugador si la respuesta es correcta o incorrecta.</li> </ol>
<b>Relaciones</b>	

## 3.4 Metodología

---

Dado que este trabajo muestra todo el proceso de elaboración de un videojuego, y uno de sus objetivos es mostrar todo el proceso de desarrollo software es necesario aplicar una metodología de desarrollo software. Entre los distintos tipos de metodologías se puede distinguir a grandes rasgos:

- Metodologías tradicionales: presentan un enfoque lineal donde cada etapa del desarrollo debe completarse secuencialmente. Se planifica el desarrollo al inicio del proyecto y se reducen al mínimo los cambios durante el proceso. La metodología tradicional más conocida y quizás mejor documentada es RUP (*Rational Unified Process*).
- Metodologías ágiles: presentan un enfoque iterativo e incremental donde según las necesidades del proyecto en el tiempo hace que los requisitos y soluciones evolucionen, es decir, permite adaptarse a los cambios que puedan surgir durante el desarrollo. Existen diferentes opciones, pero las más utilizadas son *Kanban*, *Scrum* y *Extreme Programming*.

Desde un momento inicial, debido a la naturaleza del proyecto en el que nos encontramos la metodología a llevar a cabo estaba bastante clara que iba a ser la ágil. Una vez elegida la metodología solo faltaba elegir que métodos y técnicas ágiles utilizar.

Un proyecto cuyo principal obstáculo es el tiempo que se dispone para la elaboración del mismo, uno de los obstáculos de la mayoría de los proyectos donde el tiempo es oro y hay que saber gestionarlo. Por este motivo, aunque inicialmente el objetivo era abarcar todos los contenidos del libro, estaba claro que iba a ser una tarea imposible para una sola persona. Por ello, se ha seguido un desarrollo incremental, en el que en cada iteración se han ido añadiendo distintos modos de juegos correspondientes al temario del libro.

## 3.5 Planificación

---

Una de las actividades del desarrollo software es la planificación, estimar la duración del desarrollo del proyecto. Una buena planificación es clave para estimar costes y tiempos. Además, una buena planificación en un proyecto hace que el proyecto tenga una estructura sólida y precisa siempre y cuando se sigan cada uno de los pasos y restricciones de forma correcta. La planificación conlleva estimar la duración de cada una de las tareas a desarrollar. La estimación a su vez presenta cierto grado de incertidumbre, es por ello, que esta actividad debe llevarse a cabo con detenimiento ya que cualquier fallo en la estimación puede suponer costes desmesurados en el futuro.

El desarrollo se planificó por *sprints* con una duración cada uno de un mes, empezando en febrero y teniendo como objetivo terminar a finales de mayo para tener margen de pruebas y aplicar cambios si fuera necesario.



### **Sprint 1 (febrero)**

Comienza con el desarrollo de los elementos básicos de interacción con el juego y la creación inicial de un escenario de juego: creación de unos menús muy básicos de navegabilidad entre escenas para ver la estructura del juego.

- **Creación de botones, fondos de pantalla y algunos elementos del juego:** diseño y elaboración con la herramienta Piskel de la mayoría de los elementos utilizados en el juego.
- **Pantalla instrucciones:** primera pantalla que el jugador ve cuando selecciona para jugar uno de los 10 juegos disponibles. Desde esta pantalla se pueden ver las instrucciones del juego, los mensajes de recueta y los consejos a la hora de jugar.
- **Menú de pausa:** menú disponible durante una partida que permite detener la ejecución del juego, desde el menú de pausa se puede reiniciar la partida, abandonarla, consultar el objetivo del juego, las vidas, el tiempo restante, los puntos y reanudar la partida.
- **Menú final de partida:** menú visible al finalizar una partida que permite observar las estadísticas de esta (puntos, aciertos, y duración).
- **Marcador con vidas, tiempo y puntos:** HUD del juego presente durante todo el transcurso de una partida que muestra las vidas y tiempo restantes, los puntos, incluye el botón de silenciar y el botón de pausa.

### **Sprint 2 (marzo)**

Desarrollo del bloque de juegos perteneciente al bloque de contenidos de cálculo mental: Incluye creación con Piskel de los objetos requeridos para cada juego, incluyendo frutas, personajes, animaciones y efectos. Para cada juego se desarrolla la funcionalidad del mismo y se adapta cada menú (pantalla instrucciones, menú pausa, menú final de partida) al juego.

- **Juego de sumar.**
- **Juego de restar.**
- **Juego de multiplicar.**
- **Juego de dividir.**

### **Sprint 3 (abril)**

Desarrollo del bloque de juegos perteneciente al bloque de contenidos de numeración. Al igual que el *sprint* anterior se crean y adaptan todos los recursos necesarios para cada juego.

- **Juego de mayor y menor.**
- **Juego de fracciones.**
- **Juego de par e impar.**
- **Juego de anterior o posterior.**
- **Juego de unidades, decenas y centenas.**
- **Juego de números romanos.**

### **Sprint 4 (mayo)**

Desarrollo de los distintos menús, audio, persistencia y finalización de diseños:

- **Persistencia:** creación de un sistema que permita almacenar en un fichero JSON la información de cada juego (identificador, nombre del juego, puntuación y aciertos máximos).

- **Sonidos y efectos del juego:** añadir música de fondo a los distintos menús y juegos, además de todos los efectos sonoros al pulsar botones, acertar, fallar, etc.
- **Menú de ajustes de audio:** ajustar tanto el volumen de la música de fondo como de los efectos independientemente uno del otro, utilizando *PlayerPrefs* una clase de Unity que almacena las preferencias del jugador entre distintas sesiones de juego.
- **Menú de trofeos:** menú donde se muestren todos los trofeos conseguidos y por conseguir, mostrando el requisito de cada uno para lograrlo.

## Diagrama de Gantt

En el diagrama de Gantt (figura 11) se puede apreciar la duración de cada tarea dentro de los *sprints*.



Figura 11: diagrama de Gantt

Una vez finalizado el desarrollo, surgieron algunas dudas sobre las mecánicas del juego. En el bloque de numeración, inicialmente los cuatro juegos eran de elección y ofrecían tres opciones donde una era la solución correcta. Se decidió cambiar este sistema parecido más a un examen tipo test, por un sistema donde el jugador debe hacer la operación e ir colocando uno a uno los valores del resultado, esto implica un mayor grado de atención y concentración por parte del jugador. Estos cambios se aplicaron en el mes de junio tras finalizar el juego, puesto que se vio mejor idea añadir este nuevo sistema.

## Desarrollo de un videojuego en Unity para el aprendizaje de matemáticas de tercero y cuarto curso de educación primaria

Posteriormente a la creación del videojuego se invirtió más tiempo en realizar pruebas (aparte de las pruebas realizadas durante los *sprints*), tanto del juego en sí mismo, como en niños de entre 7 y 11 años para obtener resultados que validaran la eficacia del videojuego educativo. Tras estas pruebas, en el mes de julio se aplicaron cambios moderados sobre el videojuego en el texto de algunos botones, la tipografía de la letra y el balance de la dificultad del modo multiplicación y división, pues la mayoría de los niños coincidían en que estos modos era los más difíciles.

## 4. Proceso de desarrollo

### 4.1 Arquitectura del videojuego

A continuación, se muestra un diagrama simplificado que incluye todas las clases y las relaciones más importantes para mostrar la arquitectura del videojuego (figura 12). Arquitectura formada por cinco módulos: ajustes, menús, persistencia, escenario y juegos.

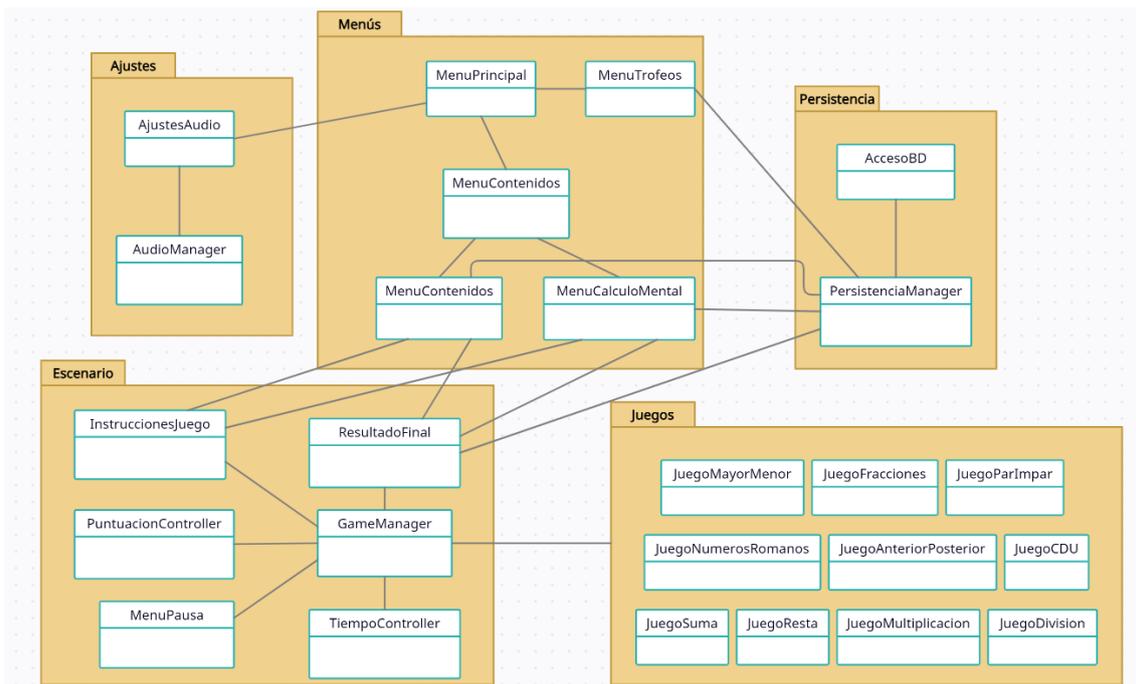


Figura 12: arquitectura del videojuego

El módulo de menús hace referencia a los distintos menús por los que se puede navegar dentro del videojuego. Cada menú es una escena distinta de Unity y cuenta con su propio script. Así pues, estos scripts se encargan de controlar el funcionamiento de cada escena.

A continuación, el módulo de ajustes, este hace referencia a la configuración de los parámetros de audio del juego. La clase `AjustesAudio` permite configurar tanto el volumen de los efectos como la música de fondo del videojuego y se comunica con la clase `AudioManager` que está permanentemente activa.

El módulo de persistencia es el encargado de guardar y cargar los datos de cada modo de juego (identificador, nombre de juego, puntuación y aciertos). La clase `PersistenciaManager` es la puerta de comunicación entre la clase `AccesoBD` y todas las demás clases que requieran de los datos almacenados.

El cuarto módulo es el escenario del juego, hace referencia a todos los componentes que interactúan durante una partida con el juego, como son las instrucciones, la puntuación,

los aciertos, el menú de pausa, etc. Todos estos objetos actúan dentro de la misma escena de Unity, en la escena de juego.

Por último, el módulo de juegos formado por las diez clases correspondientes a cada uno de los juegos. Cada clase es un juego e incluye todo lo necesario para que se pueda jugar, es decir, incluye la funcionalidad del juego en sí, las animaciones, el control de los objetos interactivables en la escena del juego, etc. Además, incluye lo necesario para comunicarse con el *GameManager* y poder acceder a los otros componentes que forman el escenario de juego.

## 4.2 Patrones aplicados

---

### 4.2.1 Singleton

---

Se ha aplicado el patrón *Singleton* de dos maneras diferentes, la tradicional y una específica de Unity. El motivo de hacerlo de la segunda forma ha sido simplemente para aprovechar la potencia de este motor. También se consiguen así dos implementaciones distintas que enriquecen este documento.

#### **Primera implementación:**

La clase *AccesoBD* se encarga de controlar la persistencia, ficheros JSON en concreto. Por seguridad y desacoplamiento es la única que lo hace. Como suele ser habitual cuando se trata de acceso a recursos, se ha implementado el patrón *Singleton* sobre ella. Así, solo existirá una instancia de la misma y los accesos a memoria serán controlados.

El diagrama UML de este patrón es muy simple (figura 13). Consta de una sola clase con una variable estática, donde guardar la única instancia de la misma que se crea durante la ejecución, y de un método público y también estático, que se encarga de devolver dicha instancia, creándola previamente si es necesario.



Figura 13: diagrama UML del patrón singleton

En la siguiente imagen (figura 14) se ve que el constructor es privado, tal y como indica el patrón. La única forma de conseguir instancias es utilizando el método *getAccesoBD*, que tiene el funcionamiento ya comentado.

```

public class AccesoBD
{
    string estadisticasJuegoRutaArchivo;
    string jsonString;

    static AccesoBD miAccesoBD;

    1 referencia
    private AccesoBD()
    {
        estadisticasJuegoRutaArchivo = Application.persistentDataPath + "/EstadisticasJuego.json";

        if (!File.Exists(estadisticasJuegoRutaArchivo))
        {
            File.WriteAllText(estadisticasJuegoRutaArchivo, "{ \"listaEstadisticas\":[{\"idJuego\":1, \"nombre\": \"JuegoResta\", \"puntuacionMax\":0, \"aciertosMax\":0}, {\"idJuego\":3, \"nombre\": \"JuegoDivisión\", \"puntuacionMax\":0, \"aciertosMax\":0}, {\"idJuego\":5, \"nombre\": \"JuegoFracciones\", \"puntuacionMax\":0, \"aciertosMax\":0}, {\"idJuego\":7, \"nombre\": \"JuegoAnteriorPosterior\", \"puntuacionMax\":0, \"aciertosMax\":0}, {\"idJuego\":9, \"nombre\": \"JuegoNúmerosRomanos\", \"puntuacionMax\":0, \"aciertosMax\":0}]}");
        }
    }

    1 referencia
    public static AccesoBD getAccesoBD()
    {
        if (miAccesoBD == null) miAccesoBD = new AccesoBD();
        return miAccesoBD;
    }
}

```

Figura 14: clase AccesoBD

A continuación (figura 15), se pueden ver los métodos que, una vez pedida la instancia de *AccesoBD*, serán llamados desde las clases que necesiten acceder a memoria.

```

public EstadisticasJuegoLista getEstadisticasJuegoLista()
{
    jsonString = File.ReadAllText(estadisticasJuegoRutaArchivo);
    return JsonUtility.FromJson<EstadisticasJuegoLista>(jsonString);
}

2 referencias
public void guardarEstadisticasJuegoLista(EstadisticasJuegoLista lista)
{
    jsonString = JsonUtility.ToJson(lista);
    File.WriteAllText(estadisticasJuegoRutaArchivo, jsonString);
}

```

Figura 15: métodos de la clase AccesoBD

Un ejemplo de ejecución se encuentra en la clase *PersistenciaManager* (figura 16).

```
static AccesoBD getAccesoBD()
{
    if (miAccesoBD == null) miAccesoBD = AccesoBD.getAccesoBD();
    return miAccesoBD;
}

1 referencia
public static void guardarPuntuacionMax(int index)
{
    EstadisticasJuegoLista lista = getAccesoBD().getEstadisticasJuegoLista();
    lista.listaEstadisticas[index].puntuacionMax = puntuacionActual;
    puntuacionActual = 0;

    getAccesoBD().guardarEstadisticasJuegoLista(lista);
}
```

Figura 16: ejemplo de uso en la clase *PersistenciaManager*

Por último, y como ya se ha comentado, utilizar el patrón *Singleton* sobre la clase *AccesoBD* logra centralizar el acceso a la persistencia y facilita así el control de esta. También es importante resaltar que, con la implementación de este patrón, se evitan problemas con los ficheros JSON. Si varias instancias de *AccesoBD* intentarían acceder a la vez podrían llegar a ocurrir conflictos graves.

### Segunda implementación:

En este caso, se ha “aplicado” el patrón *Singleton* sobre la clase *AudioManager* siguiendo una implementación típica de Unity. Aunque la finalidad es la misma, podría discutirse si llamarlo aplicación del *Singleton*. De todos modos, se trata de un ejemplo interesante para ser explicado en este documento.

La clase en cuestión es una clase con un método específico de Unity que se sobrescribe (*Awake*) para llevar a cabo el propósito de poner y cambiar la música de fondo. Debido a esto, se trata de un script que necesita ser instanciado en algún objeto de la escena. Es por eso que no se puede implementar el *Singleton* tradicional.

Como ya se ha mencionado, la tarea de esta clase es hacer funcionar la música de fondo. Esta, tiene que seguir sonando aunque la escena cambie y, como es obvio, solo debe a ver una instancia de la misma. Para llevar a cabo lo descrito se ha incluido el siguiente código el método *Awake*, que se ejecuta en el momento que se crea el objeto que contiene el *script*:

```
private void Awake()
{
    if (FindObjectsOfType<AudioManager>().Length > 1) Destroy(gameObject);

    DontDestroyOnLoad(this.gameObject);
}
```

Figura 17: método *Awake* de la clase *AudioManager*

Se observa que se busca si existe algún otro objeto en la escena que tenga este mismo *script*. En caso afirmativo, se autodestruye. De esta forma, se garantiza que únicamente existirá un único objeto con el script en todo momento, que será el primero en intentarlo.

También cabe resaltar que la llamada al método *DontDestroyOnLoad* hace que el objeto que contiene el *script* no se destruya cuando la escena cambie, justo lo contrario al comportamiento normal de Unity. Así se consigue que el objeto siga vivo y la música no pare cuando se navega entre diferentes escenas (menús del juego). Los restantes métodos de la clase simplemente se encargan de hacer funcionar la música.

Después de todo lo comentado, se aprecia que los beneficios resultantes son claros: la existencia de un solo objeto encargado de manipular la música facilita esta tarea evitando conflictos y solapes entre diferentes audios, pues se está otra vez ante un ejemplo de control de acceso a recursos.

## 4.2.2 Observador

---

Se ha aplicado el patrón *observador* para el sistema de puntuaciones máximas en los modos de juego: suma, resta, multiplicación y división, teniendo en cuenta la puntuación obtenida en cada uno de los juegos. Para llevar a cabo la implementación, por un lado, se tiene el objeto *Sujeto*, que es un objeto con los puntos obtenidos en una cierta partida y, por otro lado, se tiene los *Observadores*, que son las puntuaciones máximas conseguidas en cada uno de los cuatro modos de juegos nombrados. De esta forma, cuando los *Observadores* reciben la notificación de los puntos conseguidos por el objeto *Sujeto* se actualizan automáticamente.

El patrón ha sido adaptado a Unity, es por eso que no se ha incluido una variable booleana para comprobar si debe notificar, ya que con Unity contamos con el método *Start* (se encuentra en la clase *MenuCalculoMental*) el cual llama a *notificarObservadores()*. Este *Start* (figura 20) se ejecuta siempre al cargar la escena cuando se termina una partida en cualquiera de los cuatro modos, es por eso, que llama a *notificarObservadores()* y por tanto modifica los observadores correspondientes. Se ha tenido que utilizar la clase *MenuCalculoMental* que hereda de *MonoBehaviour* para poder utilizar el componente de texto de Unity y poder asignar las puntuaciones a los objetos de textos correspondientes de los observadores.

En este diagrama UML (figura 18) se puede observar cómo ha quedado la estructura comentada. En la figura 19 se pueden ver las implementaciones de las clases e interfaces del patrón.

Desarrollo de un videojuego en Unity para el aprendizaje de matemáticas de tercero y cuarto curso de educación primaria

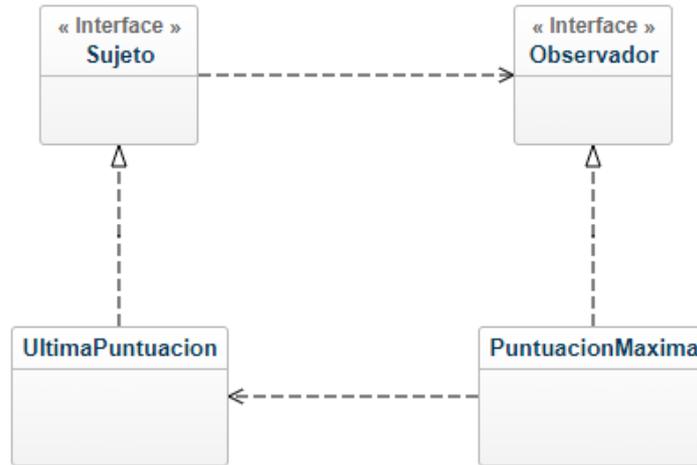


Figura 18: diagrama UML del patrón observador

```

7 referencias
public interface Sujeto
{
    5 referencias
    void añadirObservador(Observador o);
    1 referencia
    void eliminarObservador(Observador o);
    2 referencias
    void notificarObservadores();
}

13 referencias
public interface Observador
{
    2 referencias
    void actualizar();
}

1 referencia
public class UltimaPuntuacion : Sujeto
{
    private List<Observador> observadoresLista = new List<Observador>();
    5 referencias
    void Sujeto.añadirObservador(Observador o)
    {
        observadoresLista.Add(o);
    }
    1 referencia
    void Sujeto.eliminarObservador(Observador o)
    {
        observadoresLista.Remove(o);
    }
    2 referencias
    void Sujeto.notificarObservadores()
    {
        foreach (Observador observador in observadoresLista)
        {
            observador.actualizar();
        }
    }
}

5 referencias
public class PuntuacionMaxima : Observador
{
    MenuCalculoMental menuCalculoMental = GameObject.Find("GameManager").GetComponent<MenuCalculoMental>();
    private Sujeto observable = null;
    private int id;

    4 referencias
    public PuntuacionMaxima(Sujeto observable, int id)
    {
        this.observable = observable;
        this.id = id;
    }
    2 referencias
    void Observador.actualizar()
    {
        menuCalculoMental.SetPuntuacionTexto(id);
    }
}
    
```

Figura 19: clases del patrón observador

```

void Start()
{
    Sujeto ultimaPuntuacionObservable = new UltimaPuntuacion();
    Observador puntuacionObservadorSuma = new PuntuacionMaxima(ultimaPuntuacionObservable, 0);
    Observador puntuacionObservadorResta = new PuntuacionMaxima(ultimaPuntuacionObservable, 1);
    Observador puntuacionObservadorMultiplicacion = new PuntuacionMaxima(ultimaPuntuacionObservable, 2);
    Observador puntuacionObservadorDivision = new PuntuacionMaxima(ultimaPuntuacionObservable, 3);

    ultimaPuntuacionObservable.añadirObservador(puntuacionObservadorSuma);
    ultimaPuntuacionObservable.añadirObservador(puntuacionObservadorResta);
    ultimaPuntuacionObservable.añadirObservador(puntuacionObservadorMultiplicacion);
    ultimaPuntuacionObservable.añadirObservador(puntuacionObservadorDivision);

    ultimaPuntuacionObservable.notificarObservadores();
}

```

*Figura 20: método Start ubicado en la clase MenuCalculoMental*

## 4.2.3 Mediador

Otro patrón aplicado ha sido el *mediador*, que se ha aplicado para gestionar la comunicación entre los distintos componentes que forman una partida. El patrón *mediador* permite definir un objeto mediador que encapsula como un conjunto de objetos interactúan. Los objetos no se comunican entre ellos de forma directa, en su lugar se comunican haciendo uso del mediador. De esta forma se consigue reducir las dependencias entre los objetos en comunicación. Se ha seguido una estructura dinámica del patrón, formada por el mediador, los componentes y los clientes.

En el juego se cuenta con una clase llamada *GameManager* (figura 22), esta clase cumple la función de mediador. Es utilizado durante una partida para permitir la comunicación entre los distintos objetos que la forman. Por otra parte, se tienen los componentes los cuales se comunican de forma bidireccional con el mediador, son las clases *PuntuaciónController*, *ResultadoFinal* y *TiempoController*. Por último, se tiene a los clientes los cuales hacen uso del mediador para obtener la información de los componentes, los clientes son cada uno de los diez juegos implementados. A continuación se muestra la estructura seguida (figura 21).

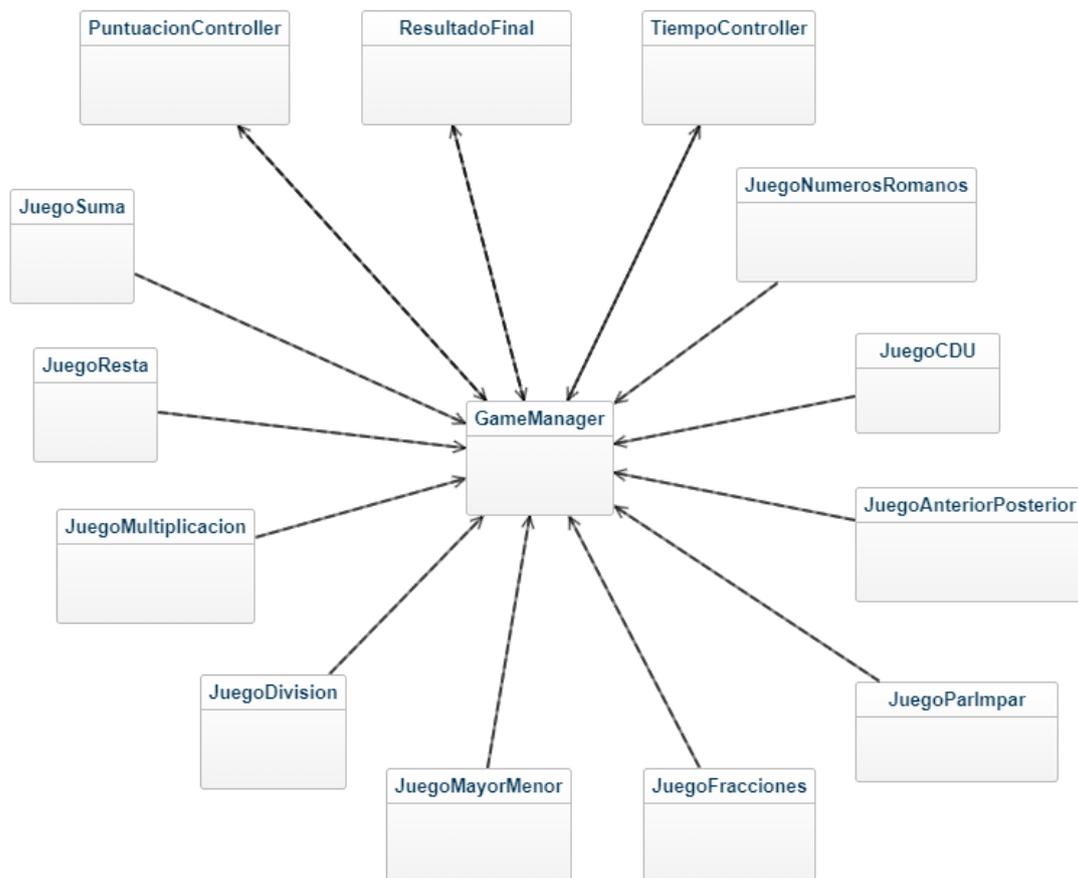


Figura 21: diagrama UML del patrón mediador

```

public class GameManager : MonoBehaviour
{
    [SerializeField] private PuntuacionController puntuacionController;
    [SerializeField] private ResultadoFinal resultadoFinal;
    [SerializeField] private TiempoController tiempoController;

    //PUNTUACIÓN
    10 referencias
    public void incrementarPuntuacion(int puntos)
    {
        puntuacionController.incrementarPuntuacion(puntos);
    }

    10 referencias
    public void decrementarPuntuacion()
    {
        puntuacionController.decrementarPuntuacion();
    }

    2 referencias
    public int getPuntuacion()
    {
        return puntuacionController.getPuntuacion();
    }

    34 referencias
    public int getAciertos()
    {
        return puntuacionController.getAciertos();
    }

    //RESULTADO FINAL
    11 referencias
    public void mostrarMenuResultadoFinal()
    {
        resultadoFinal.mostrarMenuResultadoFinal();
    }

    //TIEMPO
    10 referencias
    public void setTiempoRestante(float tiempoPartida)
    {
        tiempoController.setTiempoRestante(tiempoPartida);
    }

    10 referencias
    public void añadirTiempo(float tiempoAñadido)
    {
        tiempoController.añadirTiempo(tiempoAñadido);
    }

    11 referencias
    public void setTiempoMotonComida()
    {
        tiempoController.setTiempoMotonComida();
    }

    15 referencias
    public void setTiempoMotonComidaCorto()
    {
        tiempoController.setTiempoMotonComidaCorto();
    }

    36 referencias
    public int getEstadoMotonComida()
    {
        return tiempoController.getEstadoMotonComida();
    }
}

```

Figura 22: clase GameManager, mediador

Un programa suele estar compuesto por un número elevado de clases, cuanto mayor es la comunicación entre estas clases mayor es la dificultad de mantenimiento. Además, aplicar cualquier cambio al programa podría afectar al código de muchas otras clases. Ahora, gracias a la aplicación del patrón mediador y la centralización de un punto de comunicación con el resto de los componentes facilitará en un futuro el mantenimiento y/o refactorización del videojuego.

## 4.3 Pruebas

### 4.3.1 Pruebas generales de rendimiento

Un videojuego debe ofrecer buenos tiempos de respuesta y el menor número de pausas o menús de cargas durante el juego. Es por ello, que el rendimiento juega un papel muy importante durante el desarrollo de un videojuego. En este apartado, se ha evaluado el tiempo de carga de los menús y escenas del juego. Para ello, se ha hecho uso de las estadísticas del tiempo real de renderización de Unity.

Las pruebas han sido realizadas con un ordenador con Windows 10, Intel Core i7 2.90Ghz con gráficos integrados y 16gb de RAM. Las siguientes capturas son el resultado de la ejecución de la misma acción repetidas varias veces para contrastar posibles variaciones. Algunos menús presentaban los mismos tiempos de ejecución y carga, es por ello por lo que se han agrupado en la misma prueba para evitar una extensión innecesaria de pruebas con el mismo resultado.

Menú principal	
<b>Prueba</b>	Abrir el menú principal
<b>Prerrequisito</b>	Tener el videojuego abierto, ya que la primera vez que se inicia el videojuego tarda más en cargar el menú principal
<b>Procedimiento</b>	Acceder al menú principal desde el menú de contenidos
<b>Resultado esperado</b>	Mostrar el menú en un tiempo inferior a medio segundo
<b>Resultado obtenido</b>	El menú es mostrado en 114.19ms

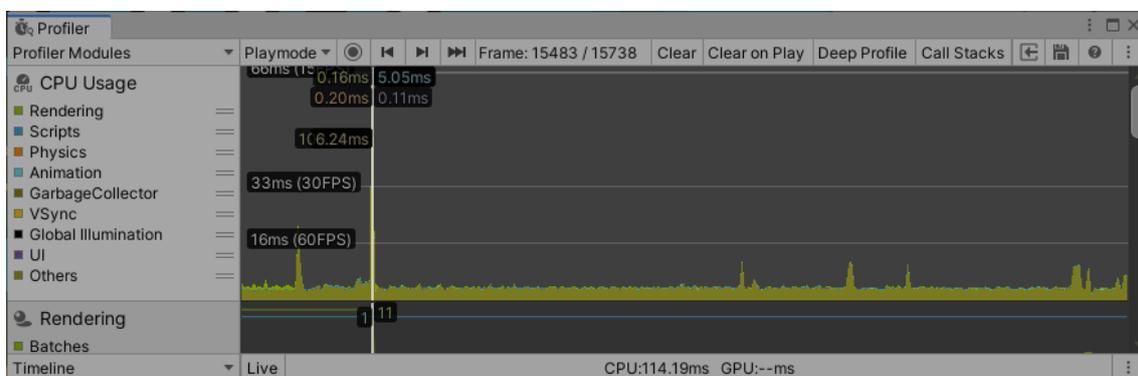


Figura 23: renderizado del menú principal

Menú trofeos y ajustes de audio	
<b>Prueba</b>	Abrir el menú de trofeos / Abrir el menú de ajustes de audio
<b>Prerrequisito</b>	
<b>Procedimiento</b>	Acceder al menú desde el menú principal
<b>Resultado esperado</b>	Mostrar el menú en un tiempo inferior a medio segundo
<b>Resultado obtenido</b>	El menú es mostrado en 159.23ms

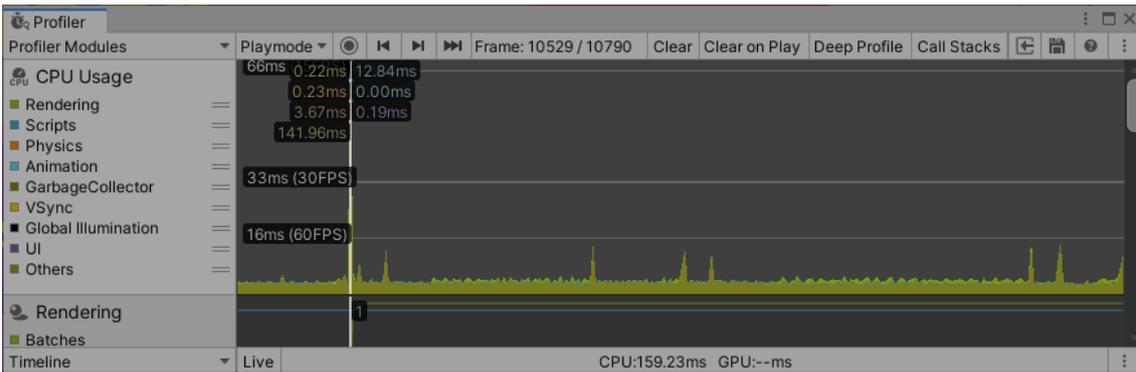


Figura 24: renderizado del menú de trofeos y del menú de ajustes de audio

Menú de contenidos y ambos bloques de contenidos	
<b>Prueba</b>	Abrir el menú de contenidos / Abrir bloques de contenidos
<b>Prerrequisito</b>	
<b>Procedimiento</b>	Acceder al menú
<b>Resultado esperado</b>	Mostrar el menú en un tiempo inferior a medio segundo
<b>Resultado obtenido</b>	El menú es mostrado en 109.28ms

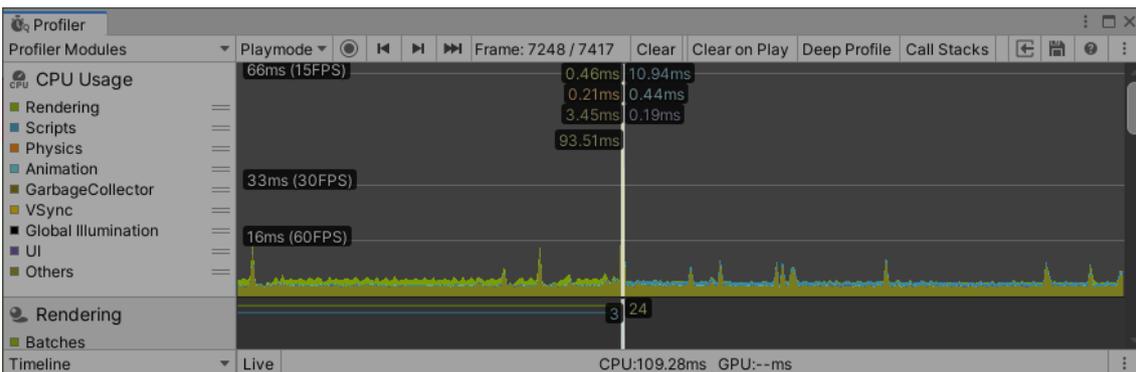


Figura 25: renderizado del menú de contenidos y de los menús de bloques de contenidos

Carga de partida	
<b>Prueba</b>	Cargar una partida
<b>Prerrequisito</b>	
<b>Procedimiento</b>	Seleccionar un modo de juego desde los bloques de contenidos
<b>Resultado esperado</b>	Mostrar el juego en un tiempo inferior a un segundo
<b>Resultado obtenido</b>	El juego es mostrado en 613.61ms

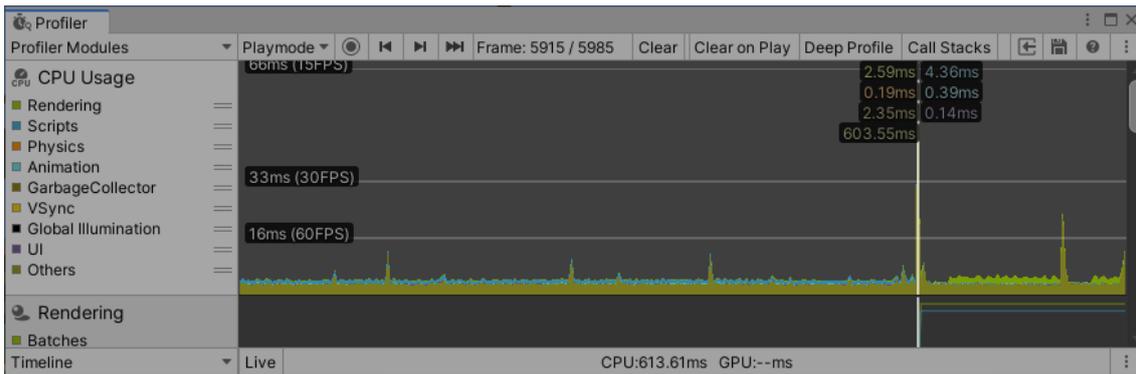


Figura 26: renderizado de una partida

Como se puede observar los resultados obtenidos son muy buenos, obteniendo tiempos de carga más que aceptables. Se observan tiempos de carga mayor y una bajada de FPS cuando se accede a un menú o se carga una partida, sin embargo, durante el resto de la ejecución del videojuego presenta unos tiempos de CPU de alrededor de 4ms y unos 200FPS (figura 27). En la siguiente captura se pueden observar estos resultados durante la ejecución del videojuego.

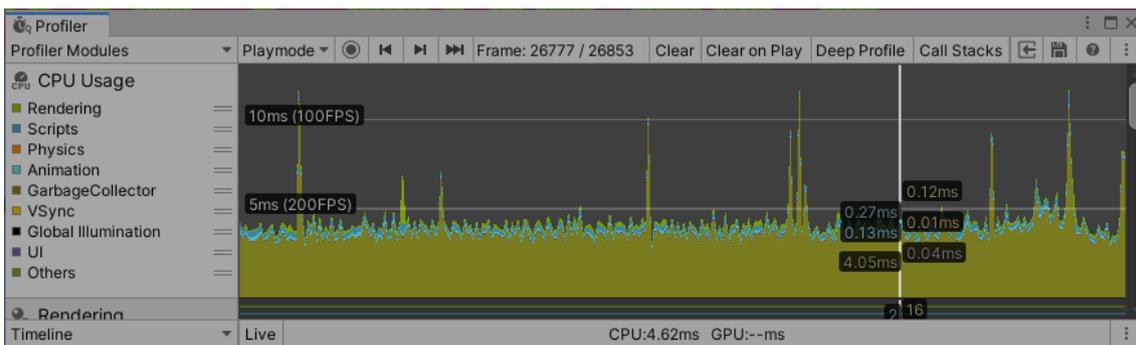


Figura 27: renderizado durante la ejecución del videojuego

## 4.3.2 Pruebas realizadas durante los sprints

---

La mayoría de las pruebas han sido realizadas durante los *sprints*, tras la implementación de cada una de las tareas descritas en el apartado de planificación. Se probaba la funcionalidad de la nueva característica implementada, buscando tanto errores de funcionamiento como de diseño. Cada nuevo modo de juego se probaba en profundidad alcanzando todas las posibilidades que se pueden dar: pasar por cada uno de los niveles de dificultad tras alcanzar cierta puntuación, comprobar todas las animaciones, tanto al ganar como al perder y las variantes de estas, agotar el tiempo, pruebas de sonido, de los controles, respuesta correcta al resultado solicitado y dificultad y contenido adecuado a la edad basándose en ejercicios similares en los libros de matemáticas nombrados en el apartado de fundamentos teóricos.

Tras la finalización de cada *sprint* el videojuego ha sido probado en ordenador, en teléfono móvil y en tableta. El ordenador cuenta con Windows 10, procesador Intel Core i7-10700 2.90GHz con gráficos integrados y 16gb de memoria RAM. El teléfono móvil es un Xiaomi pocophone f1 con procesador Snapdragon 845 con 128gb de memoria interna y 6gb de RAM. La tableta es Samsung con procesador de 1.6Ghz, con 32gb de memoria interna y 3gb de RAM. El objetivo de probar el juego en diferentes dispositivos era para ver el rendimiento en cada uno de ellos, y el ajuste al tamaño de pantalla (cada uno de ellos contaba con una resolución diferente). Este ha sido uno de los puntos muy importante durante el desarrollo, ver el videojuego correctamente en todos los dispositivos jugables.

## 4.3.3 Pruebas con niños

---

En el mes de julio se realizaron pruebas con niños de entre 7 y 11 años con el objetivo de probar la eficacia del videojuego y las sensaciones que causaba en el público. En concreto se probó el juego con siete niños y cinco niñas. Para facilitar la tarea de probar el videojuego, se instaló en un dispositivo móvil Android que se prestaba a los niños para que pudiesen jugar. Tras unos 15 o 20 minutos con ellos y ver la reacciones e interacciones con el juego se les formulaban 11 preguntas para tomar nota y obtener estadísticas que permitiesen mejorar Maths. La encuesta se realizó a través de *Google Forms*<sup>15</sup> y, para que no hubiera malentendidos, aclarar posibles dudas o se marcaran preguntas al azar sin leerlas, las preguntas se formulaban una a una a cada niño explicándoselas si era necesario. Al formular las preguntas de esta forma se obtuvo una mayor participación de los niños que no solo respondían sí o no, sino que además hacían comentarios que permitían ver realmente sus sentimientos y emoción hacia el videojuego. A continuación, se pasan a detallar los detalles de estas preguntas y los resultados obtenidos.

---

<sup>15</sup> <https://www.google.es/intl/es/forms/about/>

## 1. ¿Qué modo de juego te ha gustado más?

Al ser esta la primera pregunta, el objetivo era sencillo empezar con una pregunta fácil y rápida de responder para buscar la atención y reclamo de los niños. Esta pregunta también se formuló con la intención de clasificar los 10 juegos en un ranking de los más gustados por el público. Como puede observarse en la figura 28, el juego de par o impar es el más votado (33,3%), seguido del juego de fracciones, mayor o menor y unidades, decenas y centenas (con un 16,7%), por último, el juego de anterior o posterior y el juego de números romanos (con un 8,3%). El bloque de numeración obtiene mejores puestos que el bloque de cálculo mental y operaciones, esto se debe a que el bloque de cálculo mental y operaciones es más difícil por los juegos de multiplicación y división (resultados que se pueden observar en la pregunta 5) mientras que el bloque numeración presenta juegos más rápidos y dinámicos que atraen más a los niños.

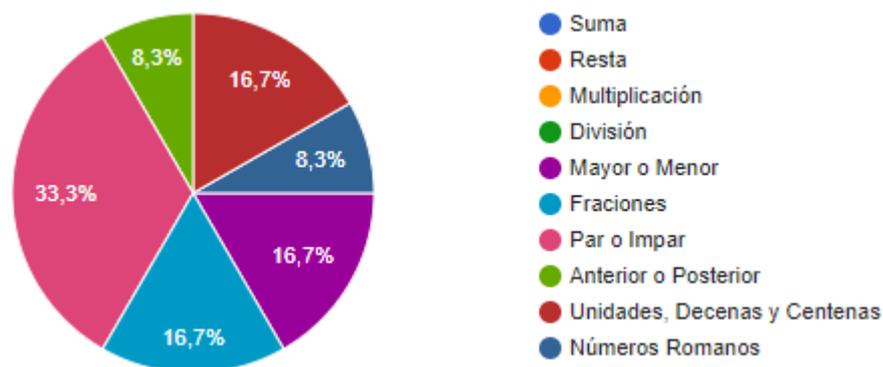


Figura 28: resultados de la pregunta 1

## 2. ¿Te ha costado entender las mecánicas de juego? Es decir ¿Cómo funcionaba y qué debías hacer en cada juego?

Al observar a los niños jugar, podía ver que algunas veces pasaban las instrucciones del juego sin leerlas o prestarles demasiada atención sin embargo pillaban las mecánicas del juego rápidamente y sabían lo que tenían que hacer sin haberlas leído. Esto acentúa que el juego es intuitivo y los jugadores localizan rápidamente los controles y el objetivo del juego. En la figura 29 se puede observar como nueve niños ha respondido que no les ha costado entender las mecánicas y otros tres que afirman haber tenido alguna duda a la hora de entender las mecánicas de juego.

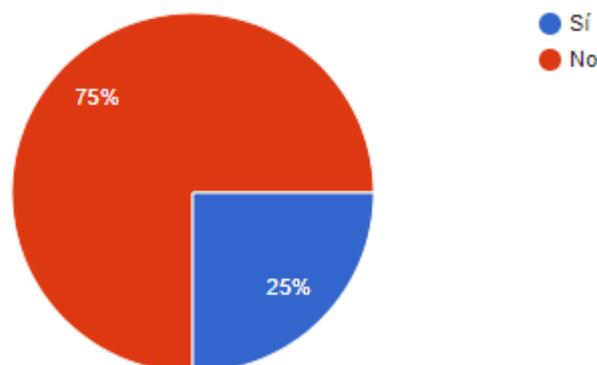


Figura 29: resultados de la pregunta 2

### 3. ¿Qué valoración le darías al arte (a las imágenes/menús) y las animaciones del videojuego?

El arte y las animaciones del videojuego era un punto débil del proyecto, ya que al no contar con un diseñador gráfico todas las imágenes, menús, botones, animaciones habían sido diseñadas por mí con una temática pixelada y aunque se hubiesen estudiado los colores y las temáticas que podrían atraer a los niños faltaba la aceptación de estos. Por suerte, los resultados fueron buenos (ver figura 30) y el trabajo de análisis realizado al principio del proyecto dio sus frutos. Se ha obtenido de media una nota de 8,42, además mientras probaban el videojuego los niños hacían comentarios positivos sobre la variedad de frutas, colores y animaciones que gratificaban el trabajo realizado, comentarios como: “Me gusta mucho la animación de las tijeras cortando la fruta” (en el juego de fracciones) o “Están guais los efectos de las puntuaciones”.

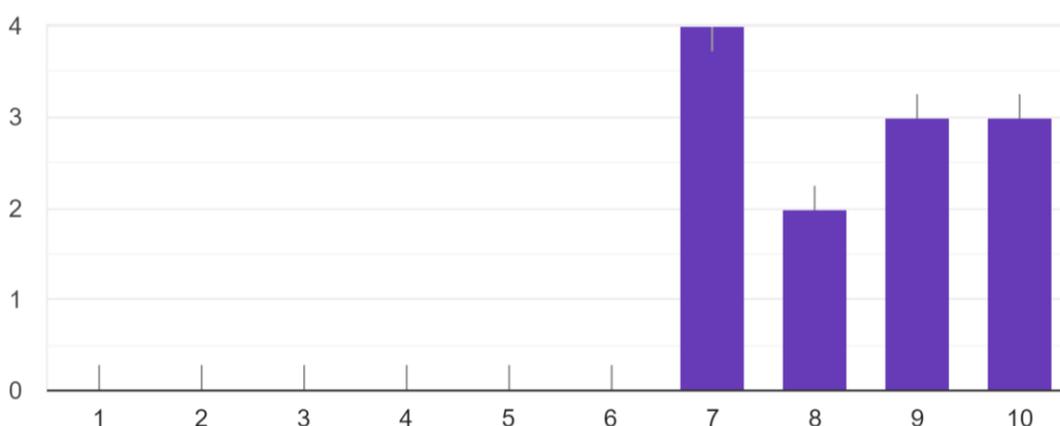


Figura 30: resultados de la pregunta 3

### 4. ¿Había algún modo especialmente difícil?

Esta pregunta se formula juntamente con la pregunta 5, tiene como objetivo destacar los juegos más difíciles y por qué lo consideran difícil. Siete niños han considerado el videojuego con un nivel de dificultad aceptable, sin embargo, cinco niños han respondido que sí había algún modo difícil. En la figura 31 pueden observarse las respuestas.

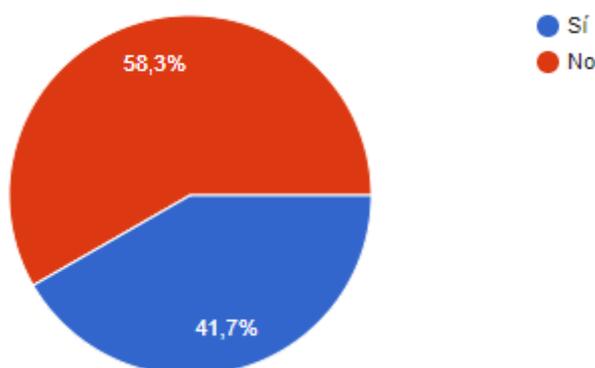


Figura 31: resultados de la pregunta 4

### 5. Si has respondido 'Sí' en la anterior pregunta, indica que modo te ha parecido más difícil. ¿Por qué lo consideras difícil?

Como se puede ver en la figura 32 los juegos más difíciles son el de multiplicar en primer lugar (60%) seguido del de dividir y sumar (20%). Al preguntar porque los consideraban difíciles las respuestas caían sobre el tiempo que tenía que invertir pensando, haciendo el cálculo en comparación con los otros modos de juego. También decían que la dificultad se incrementaba muy rápido, empezaba con multiplicaciones y divisiones sencillas, pero rápidamente subían los valores a multiplicar y necesitaban más tiempo para pensar la respuesta.

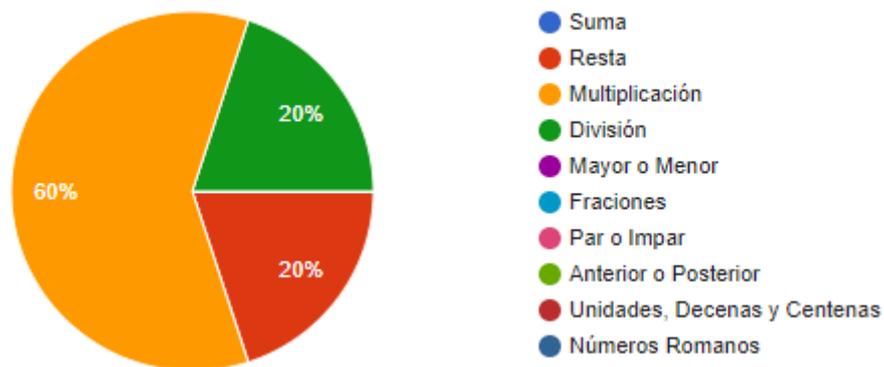


Figura 32: resultados de la pregunta 5

### 6. ¿Te ha parecido entretenido?

En esta pregunta ha habido mayoría absoluta, todos los niños que han podido jugar y probar el videojuego les ha parecido entretenido y los 20 minutos aproximadamente que probaron el videojuego se les hizo corto, queriendo jugar más. Es por ello por lo que les proporcioné a los padres el videojuego para que pudieran seguir jugando. En la figura 33 se pueden ver los resultados.

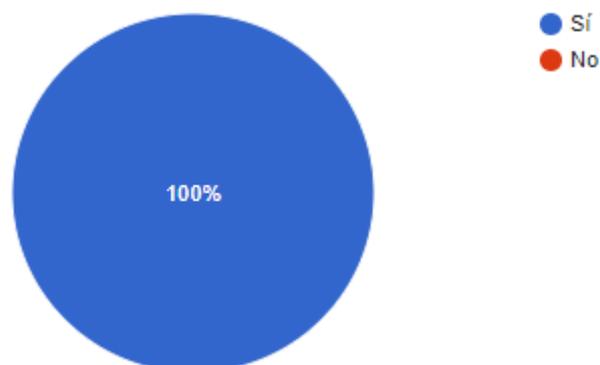


Figura 33: resultados de la pregunta 6

## 7. ¿Te gustaría más modos de juego?

Con esta pregunta se buscaba las posibilidades de ampliar el juego en futuro. Nueve niños dijeron que les gustaría más modos de juego con más frutas y personajes, añadir un modo multijugador y añadir un sistema de rankings para poder ver las puntuaciones de los amigos en cada modo de juego. Los otros tres niños les pareció bien la cantidad de modos que había, y no vieron necesario añadir más. En la figura 34 pueden observarse las respuestas de los 12 niños.

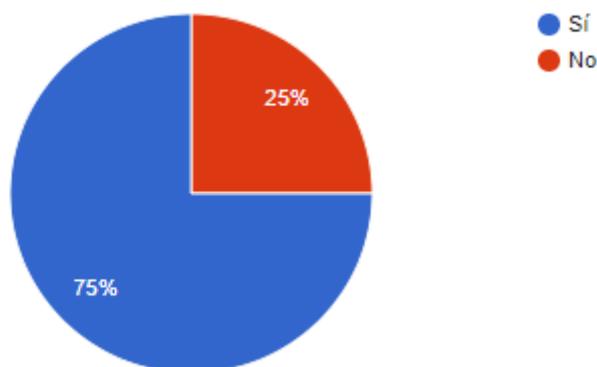


Figura 34: resultados de la pregunta 7

## 8. ¿Lo clasificarías como un videojuego o una aplicación de matemáticas?

Una de las dudas surgidas durante la fase de análisis del proyecto al analizar juegos similares, es la estrecha línea que existe a veces entre un videojuego de matemáticas y una aplicación de matemáticas. Con esta pregunta quería ver la visión desde el punto de vista de los niños, si consideraban que era un videojuego o una aplicación. Los resultados fueron mejor de lo esperado (ver figura 35) ya que 11 de los 12 niños lo clasificaron como videojuego, otra vez más todo el trabajo realizado para elaborar Maths y que no pareciera una aplicación matemática sino un videojuego tuvo sus frutos.

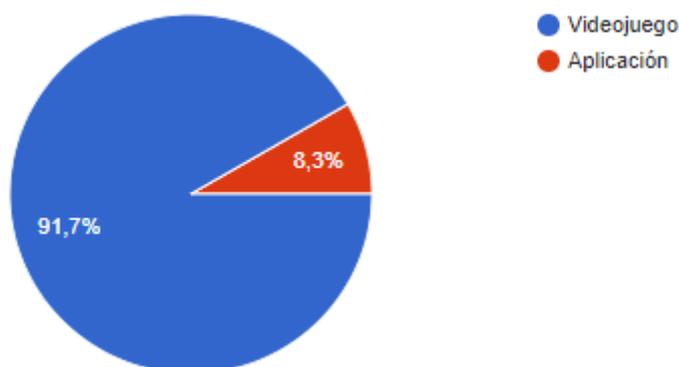


Figura 35: resultados de la pregunta 8

### 9. ¿Volverías a jugar en el futuro?

Esta preguntaba tenía como objetivo saber si volverían a jugar en un futuro para lograr todos los trofeos, alcanzar la máxima puntuación posible, probar todos los modos a fondo. La mayoría excepto uno (ver figura 36) afirmaron que sí que volverían a jugar.

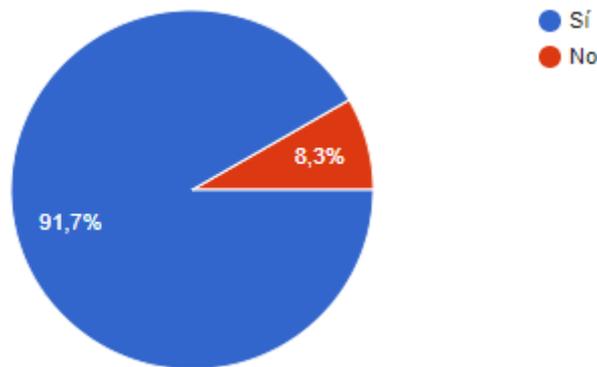


Figura 36: resultados de la pregunta 9

### 10. ¿Qué nota le pondrías a Maths?

La valoración general de Maths es buena obteniendo una nota media de 8,67 (ver figura 37) y ha sido bien recibido causando una buena percepción en los niños. Esto es un buen indicio que les guste y quieran volverlo a jugar, porque de ser así aprenderán matemáticas de una forma entretenida.

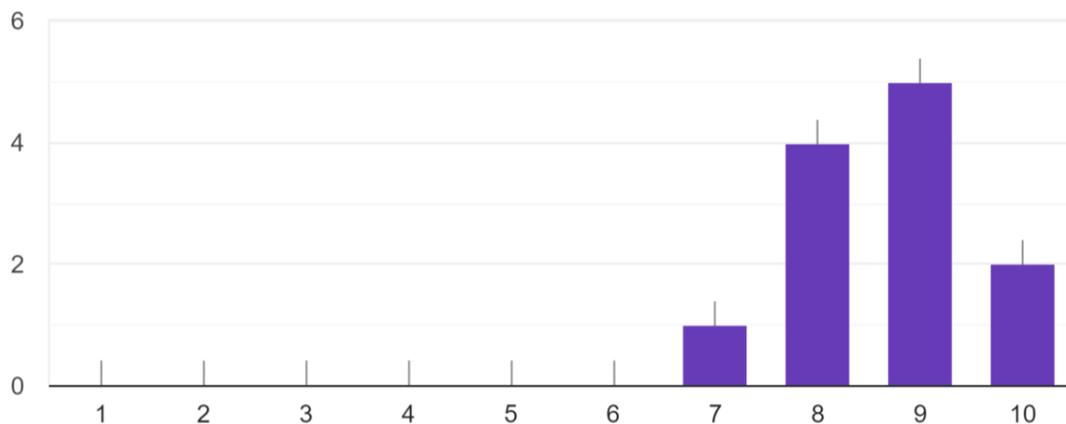


Figura 37: resultados de la pregunta 10

### 11. Sugerencias u observaciones personales. ¿Algún aspecto a mejorar?

Entre las sugerencias y aspectos a mejorar destacan la de añadir más modos, ajustar la dificultad de los juegos de multiplicar y dividir, añadir un ranking de puntuaciones para ver las puntuaciones de los otros amigos y añadir una notificación al conseguir un trofeo.

Tras probar el videojuego con niños, realizar la encuesta y analizar los resultados, Maths promete ser un videojuego con futuro. Un videojuego que con algunos pequeños cambios puede convertirse en un juego más de la biblioteca de videojuegos de los niños. Las respuestas revelan que algunas mejoras como puede ser un ranking de puntuaciones para competir con amigos podrían incrementar la motivación del público por jugar, por querer superar a los rivales y ser el primero de la lista. Algunas opiniones sobre el videojuego indicaban una dificultad de los juegos de multiplicación y división demasiado elevada, así que de cara a la versión final del juego se aplicaron cambios para incrementar más lentamente la dificultad de ambos modos de juego y así no percibir un cambio tan brusco entre una operación y la siguiente. Pese a haber una mayoría de niños indicando querer más modos de juego, el hecho de haber ya 10 modos ha sido aceptado por los niños que miraban con detenimiento qué modo elegir primero para jugar. También hay que destacar los comentarios positivos sobre las imágenes y animaciones del videojuego, comentarios que pagaban el trabajo realizado.

## 5. Tecnologías utilizadas

---

Para desarrollar el proyecto se han utilizado distintas tecnologías que pasan a describirse en los siguientes puntos indicando porque se han decidido utilizar.

### 5.1 Entorno de Desarrollo: Unity

---



*Figura 38: logo Unity*

Unity es un motor de creación de videojuegos multiplataforma creado por Unity Technologies. Unity está disponible como plataforma de desarrollo para Microsoft Windows, Mac OS y Linux. Además, la plataforma de desarrollo soporta la compilación en más de 25 tipos de plataformas. Es uno de los motores gratuitos más populares y utilizados que existen para el desarrollo de videojuegos.

La elección de Unity se debe a la experiencia con el mismo, siempre que he desarrollado un videojuego he utilizado esta herramienta principalmente por la gran cantidad de información ya sea en forma de videos, páginas web, cursos o tutoriales que existen y que permiten aprender de forma autodidacta. Una de las ventajas que tiene, es que con poco tiempo ya puede empezar a crear tu propio videojuego, la curva de aprendizaje es más corta que los demás motores como Unreal Engine<sup>16</sup> o CryEngine<sup>17</sup>. Es una herramienta fácil de aprender, pero difícil de dominar por la inmensa cantidad de opciones que ofrece.

Unity se ha elegido también porque cuenta con un sistema de repositorio integrado directamente denominado Unity Collaborate, que permite de una manera muy sencilla guardar el proyecto en la nube, almacenando distintas versiones del proyecto para poder recuperarlas en cualquier momento y desde cualquier sitio.

En Unity, los scripts se escriben en C#, un lenguaje de programación orientado a objetos que selecciona las características más relevantes de lenguajes como Visual Basic, Java o C++ y las unifica en un único lenguaje. Este lenguaje ha sido desarrollado y estandarizado por Microsoft como parte de su plataforma .NET.



*Figura 39: logo C#*

Como entorno de desarrollo para C# se ha utilizado Visual Studio un IDE que permite desarrollar aplicaciones, servicios y sitios web, aplicaciones WPF, apps de Windows Store, entre otros, para poder ser ejecutadas en todas las



*Figura 40: logo Visual Studio*

---

<sup>16</sup> <https://www.unrealengine.com/en-US/>

<sup>17</sup> <https://www.cryengine.com/>

plataformas de Microsoft Windows, Windows Mobile, Microsoft Silverlight, .NET Compact Framework y .Net Framework. Visual Studio ofrece la posibilidad de utilizar diversos lenguajes de programación. Visual Studio se puede integrar con Unity para que con un doble clic sobre un script en Unity habrá el script en Visual Studio, además los cambios realizados se aplican automáticamente a Unity y los mensajes de error en código se muestran en la consola de Unity.

## 5.2 Piskel

---



Figura 41: logo Piskel

Piskel es un editor gratuito para crear *sprites* (conjunto de imágenes combinadas en un solo archivo) animados y pixel art. Permite crear animaciones desde el navegador sin necesidad de registrarse o puedes descargar la versión de escritorio y trabajar de forma offline. Genera archivos .piskel que se pueden guardar en la nube a través de una cuenta de Gmail o en tu propio ordenador.

Se ha elegido esta herramienta por su sencillo uso, ya que permite crear imagen pintando pixeles. Por defecto viene con un tamaño de 32 por 32 pixeles, aunque puedes cambiar el tamaño. No tiene un límite establecido de pixeles, pero a partir de 500px en la versión para navegador se pueden experimentar problemas al guardar, cierres repentinos de la aplicación o ralentización. Para el proyecto se ha utilizado la versión de escritorio y la mayoría de *sprites* creados han sido de una resolución FullHD(1920x1080px) o 1024x1024px para iconos y botones. Todos los *sprites*, personajes, botones, menús, objetos de decoración han sido creados desde cero con Piskel.

Una vez creados los *sprites* se pueden exportar en formato PNG, creando una imagen única que contendrá cada uno de los cuadros que forman la animación (ver figura 42). También se puede utilizar para crear una única imagen con todos los botones que contendrá el juego o todas las cajas con distintas cantidades de fruta que se utilizan en los juegos para representar los valores de las operaciones (ver figura 43). Al importar la imagen a Unity, este permite separarla en cada uno de los cuadros que forman el *sprite* para que se pueda trabajar con ellos. Unos ejemplos creados con Piskel y utilizados en el proyecto son los siguientes:



Figura 42: sprite utilizados para crear las animaciones del frutero y la frutera

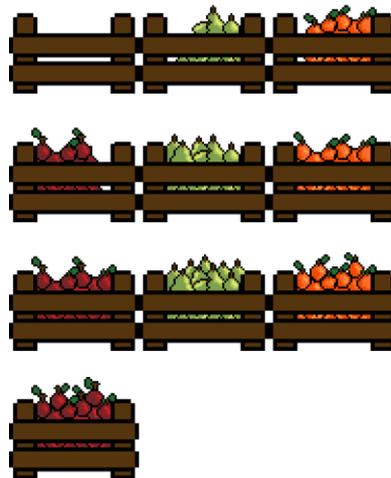


Figura 43: sprite que contiene las cajas con distintas cantidades de fruta

## 5.3 Genmymodel y Creately

---



*Figura 44: logo GenMyModel y Creately*

Tanto Genmymodel<sup>18</sup> como Creately<sup>19</sup> son plataformas web de diseño y producción de software para el modelado en línea de diagramas UML, BPMN, de flujo, modelar diagramas de clases que incluyan atributos, métodos, asociaciones, relaciones de herencia y de dependencia, etc. Han sido elegidas ambas herramientas por su experiencia de uso durante la carrera. Ambas herramientas han sido utilizadas en el proyecto para crear todos los diagramas que se pueden observar en este documento como son: el modelo de dominio, el diagrama de contexto, el diagrama de casos de uso, los diagramas UML de los patrones, la arquitectura del videojuego, entre otros. Aunque ambas herramientas contengan versiones de pago con más opciones, para el proyecto ha sido suficiente con las versiones gratuitas que han permitido elaborar todos los diagramas sin inconvenientes.

---

<sup>18</sup> <https://www.genmymodel.com/>

<sup>19</sup> <https://creately.com/es/home/>

## 6. Conclusiones

---

Una vez finalizado el trabajo y llegados a este punto, cabe destacar en primer lugar el cumplimiento de todos los objetivos propuestos a principios de iniciarse en el desarrollo de un videojuego en Unity para el aprendizaje de matemáticas de tercero y cuarto curso de educación primaria, recorriendo todas las etapas del desarrollo software hasta elaborar un videojuego educativo de matemáticas. En concreto se ha pasado por las etapas de: planificación, análisis, diseño, implementación, pruebas e instalación. Para las pruebas con niños, en un principio se veía como una tarea complicada por la pandemia, sin embargo, finalmente se pudo probar el videojuego con 12 niños. Para lograr el segundo objetivo se quería contar con la colaboración por parte de alguna editorial para tener acceso a libros de texto, pero, pese a no tener la colaboración de ninguna editorial se pudo acceder a libros de primaria para extraer los contenidos matemáticos en lo que se basa Maths. También se ha logrado el objetivo de analizar videojuegos educativos similares, para ello, se han probado videojuegos actuales de matemáticas a través de sitios web y de *Google Play*<sup>20</sup>, realizando un análisis comparativo para extraer las características implementadas en Maths. En cuanto al estudio de las metodologías educativas para enseñar a través de los videojuegos, también se considera un objetivo alcanzado ya que se han estudiado: libros de matemáticas de tercero y cuarto de primaria, la gamificación, el canal de flujo, la teoría del aprendizaje de Piaget, los colores y su significado. Todos estos objetivos han hecho posible alcanzar el último objetivo, crear un videojuego de matemáticas educativo, Maths. Para llevar a cabo todo el desarrollo se tuvieron que reforzar los conocimientos con Unity y C#, aprender a construir las interfaces y animaciones con Piskel y utilizar Genmymodel y Creately para el modelado.

A nivel personal, este proyecto ha sido de gran utilidad en mi continua formación con Unity, ya que he tenido que consumir gran cantidad de tutoriales para aprender funcionalidades desconocidas pero necesarias para el desarrollo del videojuego. También me ha permitido enfrentarme cara a cara con el proceso de desarrollo software, con los retos que supone, las adversidades a las que te enfrentas, explorando diferentes soluciones y eligiendo la más adecuada. Aunque no ha sido fácil el desarrollo de un videojuego en solitario, donde una única persona debía cubrir todos los campos del desarrollo, estoy muy contento con los resultados obtenidos, puesto que finalmente el juego cumple las expectativas iniciales obteniendo buenas vibraciones en las pruebas realizadas con niños. No me arrepiento de haber desarrollado este proyecto pues ha resultado una constante montaña de sabiduría en la que cada día he aprendido algo nuevo. Puedo concluir afirmando que este proyecto me ha permitido poner en práctica todo lo aprendido durante la carrera, especialmente poner en práctica los conocimientos aprendidos en la rama de software, en asignaturas como *Proceso del Software*, *Proyecto de Ingeniería del Software* o *Diseño de Software* y también la optativa de *Introducción a la Programación de Videojuegos* gracias a la cual surgió la idea de desarrollar un videojuego como trabajo de final de grado.

---

<sup>20</sup> <https://play.google.com/store>

## 7. Trabajo futuro

---

El juego ha sido exportado tanto para Windows como para Android ambos archivos de descarga se encuentra en el repositorio [github.com/GCGabriel](https://github.com/GCGabriel)<sup>21</sup> de forma pública para ser descargados por cualquier persona de manera gratuita. De forma que el videojuego esté disponible para todos aquellos padres y madres que busquen un juego de matemáticas educativo para sus hijos, o para aquellos curiosos que quieran entretenerse y comerse la cabeza con unas partidas. En el repositorio también se encuentra el proyecto de Unity completo con la finalidad de que el público pueda descargárselo y añadir nuevos contenidos como pueden ser los demás cursos de primaria, más modos de juego, o completar los bloques de resolución de problemas, medida y geometría y tratamiento de la información que faltan de tercero y cuarto. Los tres bloques no implementados por falta de tiempo dotarían al videojuego del contenido completo de ambos cursos. También como resultado de la encuesta realizada con niños se podría añadir un ranking de puntuaciones por cada modo de juego donde se mostrará la puntuación de los jugadores y así fomentar la competitividad del videojuego.

La finalidad de hacer público los archivos de desarrollo del videojuego, es para que no quede en el olvido ya que desde la experiencia de uso de personas que lo han jugado y bajo mi punto de vista es un juego con potencial para la educación. El cual se podría mejorar e incrementar en contenidos por la comunidad de desarrolladores de videojuegos en Unity, o incluso exportar la idea a otro motor gráfico con mayores ventajas.

---

<sup>21</sup> <https://github.com/GCGabriel>



## Referencias y bibliografía

---

- [1]. Coll, César. *Aprender y enseñar con las TIC: expectativas, realidad y potencialidades*. Boletín de la institución libre de enseñanza. 2008. págs. 7-40. Vol. 72 N° 1.
- [2]. Gómez Martín, Pedro Pablo, González Calero, Pedro Antonio y Gómez-Martín, Marco A. *Aprendizaje basado en juegos*. Icono14. 2004. págs. 2-3. Vol. 2 N° 2. ISSN: 1697-8293.
- [3]. M. Esther del Moral Pérez, Alba P. Guzmán Duque, L. Carlota Fernández García. *Aprendizaje basado en juegos: activando las inteligencias lógico-matemáticas, naturalista y lingüística en el alumnado de Primaria*. Píxel-Bit. Julio 2016. págs. 177-193. Vol. 7 N°49. ISSN:1133-8482.
- [4]. García, D.M, Joaquín, M. y Torres, P. y Vázquez, I.R. *Estilos de enseñanza y las nuevas tecnologías en la educación*. ReiDoCrea. 2013. págs. 219-225. Vol. N° 2. ISSN: 2254-5883.
- [5]. Flórez, Marcela, y otros. *Sociedad del conocimiento, las TIC y su influencia en la educación*. Espacios. 2017. pág. 39. Vol. 38 N° 35. ISSN: 0798-1015.
- [6]. Melo Herrera, Mónica Patricia. *Videojuegos y ocio: algunas reflexiones sobre su incidencia en el desarrollo físico-social de los jóvenes*. Lúdica pedagógica. 2016. págs. 87-91. Vol. 1 N° 23. ISSN: 0121-4128.
- [7]. Contreras, Ruth Sofhía. *Acercamiento a las características de los videojuegos y sus beneficios en el aprendizaje*. 2013. págs. 381-394. ISBN 978-84-686-4895-8.
- [8]. Asociación Española de Distribuidores y Editores de Software de Entretenimiento, (aDeSe). *Anuario de la industria del videojuego - El videojuego en las aulas*. aDeSe. 2012. págs. 30-39.
- [9]. Gros Salvat, Begoña y Garrido Mirando, José María. *"Con el dedo en la pantalla": el uso de un videojuego de estrategia en la mediación de aprendizajes curriculares*. Ediciones Universidad de Salamanca. Noviembre 2008. págs. 108-129. Vol. 9 N° 3. ISSN: 1138-9737.
- [10]. Albarracín, Lluís, Hernández-Sabaté, Aura y Gorgorió, Núria. *Los videojuegos como objeto de investigación incipiente en Educación Matemática*. Modelling in Science Education and Learning. 2017. págs. 53-72. Vol. 10 N° 1. ISSN: 1988-3145.
- [11]. Valcárcel Nazco, Cristina. *El videojuego como recurso didáctico en el aprendizaje de las matemáticas en primer curso de Educación Secundaria Obligatoria*. Universidad Internacional de La Rioja. Octubre 2013. págs. 30-34.
- [12]. Pareto, Lena, y otros. *A teachable-agent-based game affording collaboration and competition: Evaluating math comprehension and motivation*. s.l. : Educational Technology Research and Development, 2012. págs. 723-751. Vol. 60 N° 5. DOI: 10.1007/s11423-012-9246-5.

- [13]. Shuler, Carly. *What in the world happened to carmen sandiego? The edutainment era: Debunking myths and sharing lessons learned*. The Joan Ganz Cooney Center. 2012. págs. 4-22.
- [14]. Díez-Gutiérrez, Enrique-Javier. *Gobernanza híbrida digital y Capitalismo EdTech: la crisis del COVID-19 como amenaza*. Foro de Educación. 2021. págs. 105-133. Vol. 19 N° 1. ISSN: 1698-7802.
- [15]. Piaget, Jean. *La teoría de Piaget*. Infancia y aprendizaje. 1981. págs. 13-54. Vol. 4 (sup2). DOI: 10.1080/02103702.1981.10821902.
- [16]. Real Academia Española. (s.f.). *Gamificación*. En Diccionario de la lengua española. Recuperado en 18 de julio de 2021, de <https://www.rae.es/observatorio-de-palabras/gamificacion>.
- [17]. Gaitán, Virginia. *Gamificación: el aprendizaje divertido*. educativa. 2013. Vol. 15.
- [18]. Schell, Jesse. *The Art of Game Design: A book of lenses*. CRC Press. 2008. págs. 118-123. ISBN: 978-0-12-369496-6.
- [19]. Csikszentmihalyi, Mihaly. *Fluir (flow): una psicología de la felicidad*. Kairós. 2010. pág. 75. ISBN-13: 978-84-7245-372-2.
- [20]. Barros, Rafael José, Rojas, John Alexander y Sánchez, Luz Marina. *Diseño de instrumentos didácticos para aprendizaje activo basado en teoría de colores*. Revista Educación en Ingeniería. 2008. págs. 11-18. Vol. 3 N° 5. ISSN: 1900-8260.

# Índice de ilustraciones

Figura 1: profesores que han utilizado videojuegos en la educación .....	13
Figura 2: tamaño del mercado de software y aplicaciones educativas en todo el mundo de 2019 a 2024 (en millones de dólares estadounidenses), Statista.....	15
Figura 3: captura de dos escenarios distintos de Monster Numbers .....	16
Figura 4: captura de tres juegos de Mundoprimeria.....	17
Figura 5: capturas del juego Juegos matemáticos.....	18
Figura 6: portada de los libros de matemáticas utilizados.....	22
Figura 7: concepto del canal de flujo (19).....	24
Figura 8: diagrama de dominio .....	35
Figura 9: diagrama de contexto.....	35
Figura 10: diagrama de casos de usos .....	37
Figura 11: diagrama de Gantt .....	43
Figura 12: arquitectura del videojuego.....	45
Figura 13: diagrama UML del patrón singleton .....	46
Figura 14: clase AccesoBD .....	47
Figura 15: métodos de la clase AccesoBD.....	47
Figura 16: ejemplo de uso en la clase PersistenciaManager .....	48
Figura 17: método Awake de la clase AudioManager .....	48
Figura 18: diagrama UML del patrón observador.....	50
Figura 19: clases del patrón observador .....	50
Figura 20: método Start ubicado en la clase MenuCalculoMental .....	51
Figura 21: diagrama UML del patrón mediador .....	52
Figura 22: clase GameManager, mediador .....	53
Figura 23: renderizado del menú principal.....	54
Figura 24: renderizado del menú de trofeos y del menú de ajustes de audio .....	55
Figura 25: renderizado del menú de contenidos y de los menús de bloques de contenidos .....	55
Figura 26: renderizado de una partida.....	56
Figura 27: renderizado durante la ejecución del videojuego .....	56
Figura 28: resultados de la pregunta 1 .....	58
Figura 29: resultados de la pregunta 2.....	58
Figura 30: resultados de la pregunta 3.....	59
Figura 31: resultados de la pregunta 4 .....	59
Figura 32: resultados de la pregunta 5.....	60
Figura 33: resultados de la pregunta 6.....	60
Figura 34: resultados de la pregunta 7.....	61
Figura 35: resultados de la pregunta 8.....	61
Figura 36: resultados de la pregunta 9.....	62
Figura 37: resultados de la pregunta 10 .....	62
Figura 38: logo Unity.....	64
Figura 39: logo C# .....	64
Figura 40: logo Visual Studio .....	64
Figura 41: logo Piskel.....	65
Figura 42: sprite utilizados para crear las animaciones del frutero y la frutera .....	66
Figura 43: sprite que contiene las cajas con distintas cantidades de fruta .....	66
Figura 44: logo GenMyModel y Creately.....	67

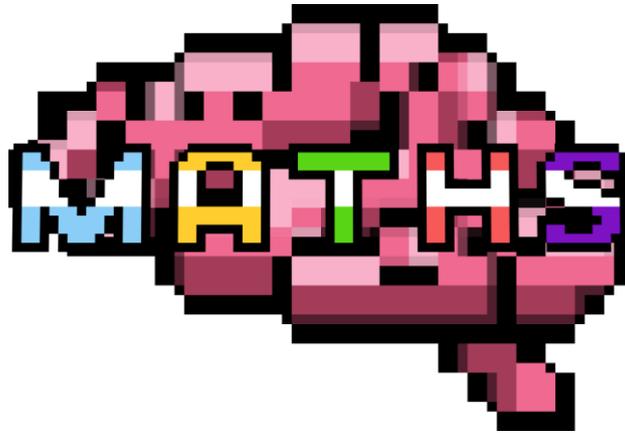
# Definiciones, abreviaturas y acrónimos

---

- **BPMN:** Business Process Model and Notation (Modelo y Notación de Procesos de Negocio).
- **CPU:** Central Processing Unit (Unidad Central de Procesamiento).
- **FPS:** Frames per second (Fotogramas por segundo).
- **HUD:** Head-Up Display. Conjunto de la información que se muestra en el juego mediante iconos, números o mapas, ya sea sobre el estado de nuestra partida y/o nuestro personaje.
- **IDE:** Integrated Development Environment (Entorno de desarrollo integrado).
- **IDR:** Ingeniería de Requisitos.
- **JSON:** JavaScript Object Notation. Formato de texto sencillo para el intercambio de datos.
- **PlayerPrefs:** clase de Unity que almacena las preferencias del jugador entre distintas sesiones de juego.
- **PNG:** Portable Network Graphics. Es un formato gráfico.
- **RAM:** Random Access Memory (Memoria de Acceso Aleatorio).
- **RUP:** Rational Unified Process. Es un proceso de desarrollo de software.
- **Scrum:** marco de trabajo para desarrollo ágil de software.
- **Sprints:** nombre que recibe cada uno de los ciclos o iteraciones que existen dentro de un proyecto Scrum.
- **Stakeholders:** son los diferentes grupos de personas interesadas en el proyecto a desarrollar y que influyen en el desarrollo de este.
- **TFG:** Trabajo final de grado.
- **TIC:** Tecnologías de la información y la comunicación.
- **UML:** Unified Modeling Language (Lenguaje unificado de modelado).
- **WPF:** Windows Presentation Foundation. Permite el desarrollo de interfaces de interacción en Windows tomando características de aplicaciones Windows y de aplicaciones web.

## Anexo I: documento de diseño

---



*Ilustración 1: logo de Math*

### Versión del documento de diseño y fecha de la última revisión

- 1 (16/05/2021), Gabriel Caravaca Gosp

### Análisis del juego

Maths es un videojuego en el que con la ayuda del intelecto y las matemáticas deberás superar distintos retos matemáticos. Como jugador deberás hacer uso de tus habilidades en aula para aplicarlas al videojuego y lograr la mayor cantidad de puntos y aciertos posibles.

### Declaración de misión

Maths es un nuevo juego arcade con temática retro adaptado a las nuevas generaciones para dispositivo móvil.

### Género

Videojuego educativo

Arcade

### Plataformas

Android

PC

## Público objetivo

Dirigido principalmente a niños de entre 8 y 10 años correspondientes a los cursos de tercero y cuarto de primaria, para el cual han sido diseñados los contenidos. Sin embargo, dentro de la categoría PEGI se clasifica en PEGI 3. Es un videojuego adecuado para todas las edades, el cual no contiene ni palabras malsonantes, ni contenidos que puedan asustar a los niños pequeños. Sus contenidos y retos son aptos para todo aquel que le gusten las matemáticas.

## Descripción general

Maths se divide en 10 minijuegos diferentes, todos ellos combinan las matemáticas con la fruta. Cada minijuego corresponde con un tema del libro de matemáticas: sumas, restas, divisiones, multiplicaciones, mayor y menor, par e impar, fracciones, números romanos, etc. Además, cada minijuego tiene un objetivo diferente pero siempre relacionado con ayudar a los fruteros y fruterías con la fruta, a contarla, transportarla, cargarla, a indicarles donde hay más, entre otros objetivos.

## Jugabilidad

### Resumen del juego

El juego aborda las matemáticas desde un punto de vista práctico y enriquecido a nivel visual. Incluye contenidos matemáticos sobre numeración, operaciones y cálculo mental, los bloques más importantes a nivel escolar y extraescolar. La jugabilidad se basa principalmente en hacer uso del conocimiento y la lógica para resolver los distintos problemas matemáticos propuestos. A mayor conocimiento mayor será la puntuación obtenida en cada uno de los juegos, así que cuanto más se juegue, mejor puntuación se obtendrá y mayores conocimientos se adquirirán.

### Experiencia del jugador

Como jugador te verás inmerso en una aventura para ayudar a los fruteros con sus distintas tareas haciendo uso de tu intelecto matemático, sentirás que las matemáticas no son solo una asignatura teórica de números, verás que tienen su utilidad en el mundo que nos rodea. Te adentrarás en los conocimientos del libro matemático, en sus temas, en sus operaciones y propiedades, recorrerás cada parte de libro y te enfrentarás a los distintos problemas que nos plantea.

### Objetivos y recompensas del juego

Cada uno de los 10 juegos cuenta con un objetivo. La recompensa dentro del videojuego es la puntuación y cantidad de aciertos conseguidos en cada juego, además de los trofeos obtenidos como resultado de alcanzar cierta puntuación o número de aciertos. A nivel interior y personal la recompensa es el aprendizaje. De una forma sencilla y entretenida como es un videojuego se pueden adquirir conocimientos, habilidades y valores.

## Diseño de escenarios

El videojuego consta de 10 juegos cada uno de ellos con un escenario.

### Suma

El juego de sumar consiste principalmente como su nombre indica en realizar sumas. Dentro del videojuego el objetivo es salvar a las personas del dragón, para ello hay que alimentar al dragón con una cantidad de comida exacta, esta cantidad exacta es el resultado de la suma. Los dos valores para sumar se muestran dentro de dos cajas de fruta. El resultado que debe introducir el jugador, debe hacerlo en la zona redondeada de la imagen de abajo, colocando las centenas, decenas y unidades, una vez tenga la solución debe pulsar la flecha de la derecha para comprobar la solución. El valor de la suma puede oscilar entre 10 y 999. Una vez acierte la respuesta se pasa a la siguiente operación.

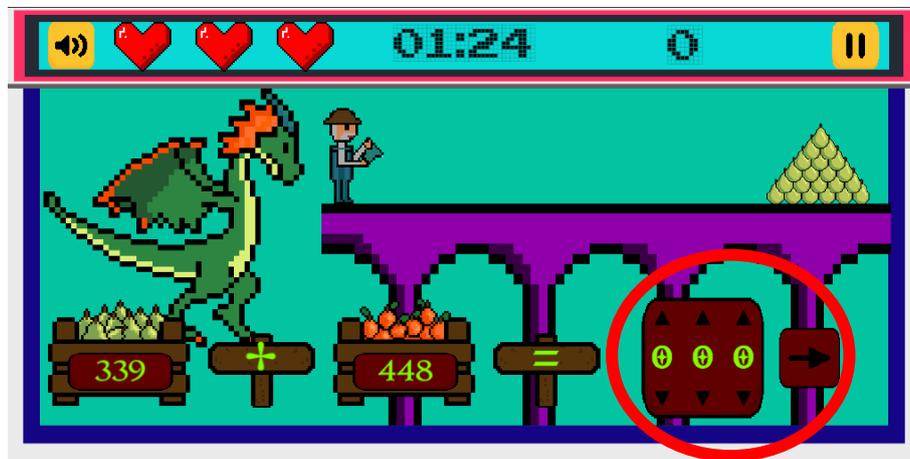


Ilustración 2: escenario del juego suma

### Resta

El juego de resta es idéntico al de sumar, salvo que las operaciones a resolver en este caso son restas.

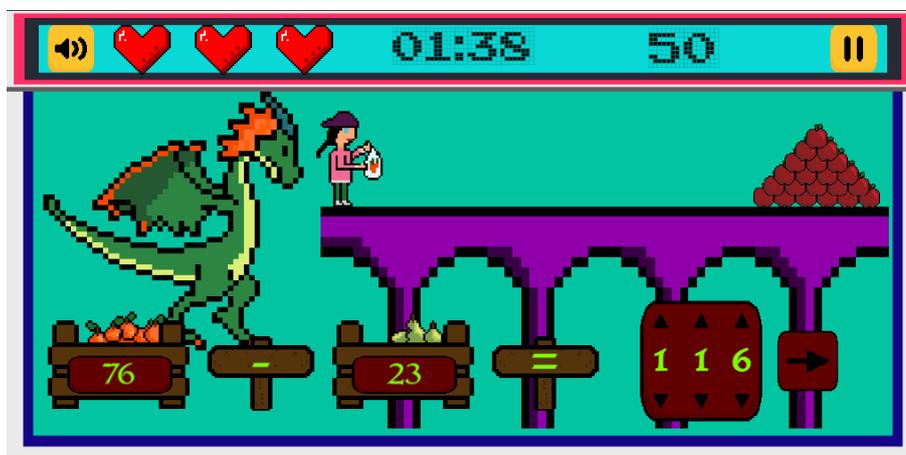


Ilustración 3: escenario del juego resta

## Multiplicación

El juego de multiplicar tiene por objetivo ayudar al frutero a contar la cantidad total de fruta que tiene, para ello hay que multiplicar la cantidad de fruta que tiene cada caja (multiplicando) por el número de cajas (multiplicador). El enunciado de la multiplicación se muestra en el centro de la pantalla, además el número de cajas visibles varía en función del multiplicador. El resultado, al igual que en los dos juegos anteriores se debe colocar en la parte de abajo y una vez colocado comprobarlo. La dificultad viene definida por el valor del multiplicando (inicialmente oscila entre 1 y 300) y el valor del multiplicador (inicialmente oscila entre 1 y 4). Cada 10 aciertos se incrementan ambos valores, hasta alcanzar un rango máximo entre 1 y 999 para el multiplicando y un rango entre 1 y 10 para el multiplicador tras alcanzar 40 aciertos.

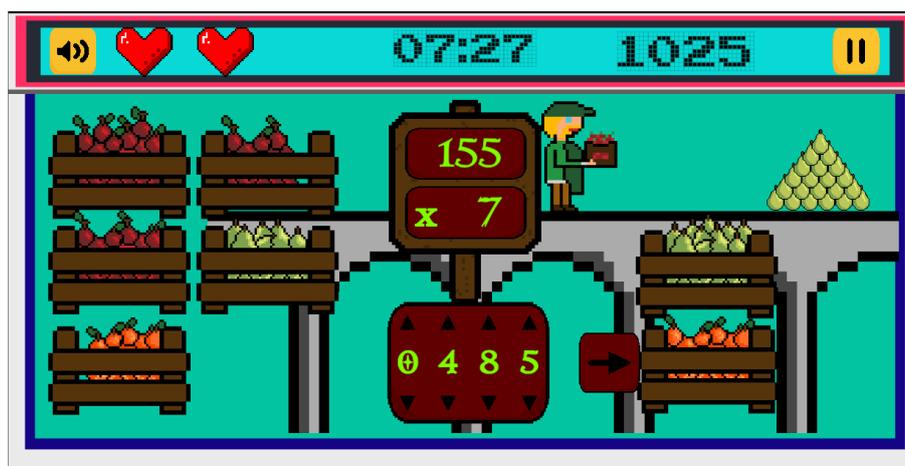
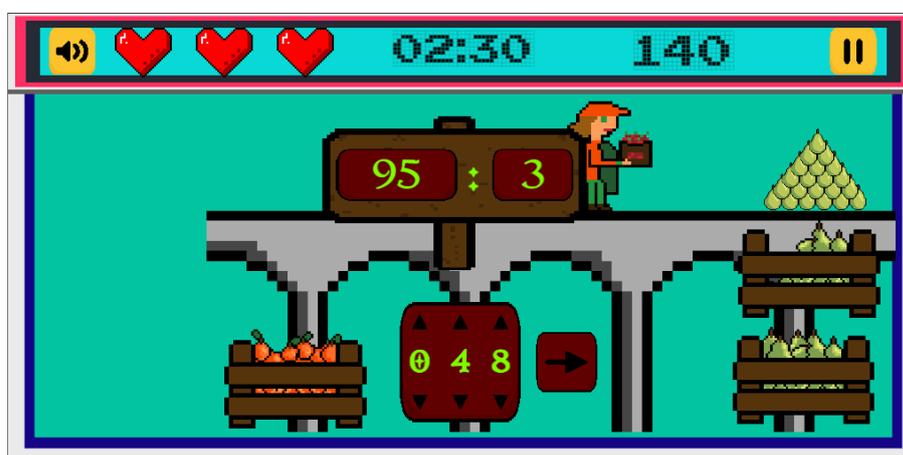


Ilustración 4: escenario del juego multiplicación

## División

El juego de dividir tiene por objetivo ayudar al frutero a repartir la fruta entre las cajas que tiene disponibles, para ello hay que dividir la cantidad de fruta que tiene (dividendo) entre el número de cajas (divisor). El enunciado de la división se muestra en el centro de la pantalla, y al igual que el juego de multiplicar, las cajas visibles varían en función del valor de divisor. La solución se debe colocar en la parte de abajo y una vez colocado comprobarlo. La dificultad viene definida por el valor del dividendo (inicialmente oscila entre 1 y 300) y el valor del divisor (inicialmente oscila entre 1 y 4). Cada 10 aciertos se incrementan ambos valores, hasta alcanzar un rango máximo entre 1 y 999 para el dividendo y un



dividendo y un rango entre 1 y 10 para el divisor tras alcanzar 40 aciertos.

Ilustración 5: escenario del juego división

## Mayor y menor

El juego de mayor y menor tiene por fin ayudar a los fruteros a cargar las cajas que más fruta tienen en la camioneta, para ello el jugador debe seleccionar el signo correcto (de entre los tres disponibles, botones de abajo) de forma que los fruteros sepan en que caja hay más fruta. La dificultad avanza a medida que se consiguen aciertos, inicialmente los números mostrados como enunciado están formados por tres cifras. Al llegar a 20 aciertos, los números pasan a tener 4 cifras, cuando se logran 40 aciertos pasan a tener 5 cifras, a los 60 aciertos vuelven a ser 3 cifras, pero esta vez con un decimal y el llegar a 80 aciertos se alcanza la dificultad máxima con 4 cifras y un decimal.

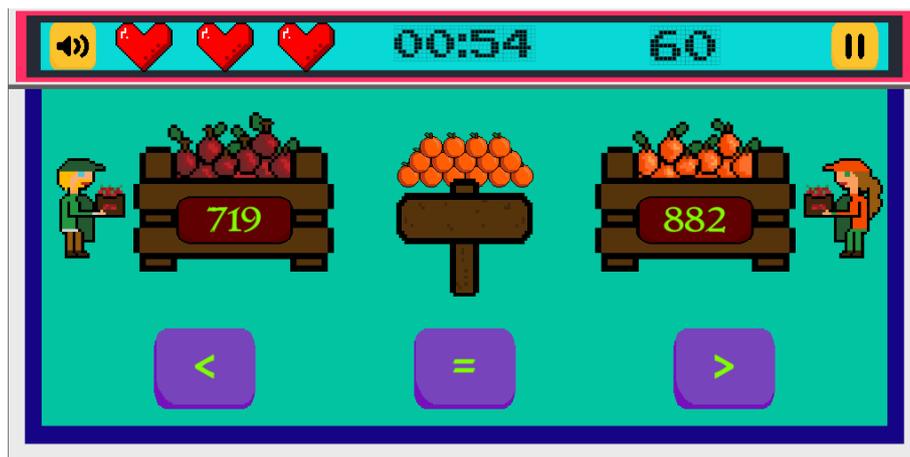


Ilustración 6: escenario del juego mayor y menor

## Fracciones

El juego de fracciones tiene como objetivo identificar la porción de fruta representada, para que el fruto sepa la porción que debe repartir a sus clientes. Se ofrecen tres soluciones posibles para la fracción mostrada, pero solo una es correcta. Inicialmente las porciones de fruta mostradas vienen delimitadas por un borde negro que indica en cuantos trozos está dividida la pieza. Al llegar a 40 aciertos estas rayas delimitadoras desaparecen aumentando la dificultad de juego ya que el jugador tiene que pesar cual es el denominador y numerador simplemente viendo la porción de fruta coloreada.

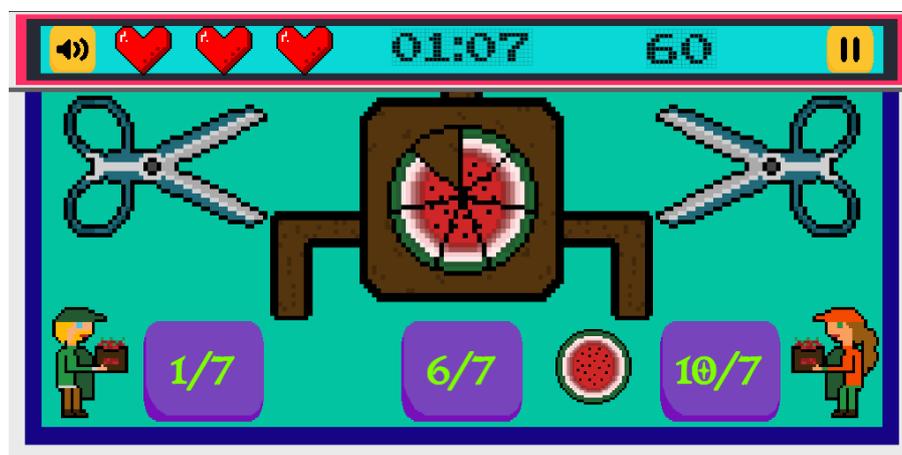


Ilustración 7: escenario del juego fracciones

## Par o Impar

El juego de par o impar tiene por fin indicar a los frutereros si deben cargar una cantidad par o impar de fruta en la camioneta, para ello el jugador debe contar el número de frutas que hay en el juego y decir si es una cantidad par o impar. La dificultad viene definida por la cantidad de frutas que se muestran, a mayor cantidad mayor tiempo invertido en contarlas y mayor posibilidad de error. Inicialmente la cantidad mostrada oscila entre 1 y 5 frutas, cada cinco aciertos este margen aumenta hasta llegar a 30 frutas, margen que se alcanza a lograr 60 aciertos.

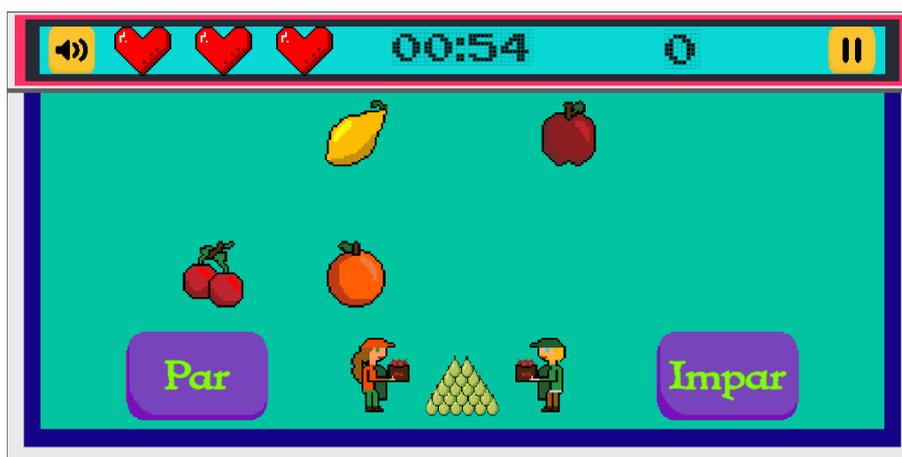


Ilustración 8: escenario del juego par o impar

## Anterior o posterior

El objetivo del juego es indicar a los frutereros si deben llevar la fruta a la tienda anterior o posterior. En este juego, se muestra como enunciado un valor y se pide el valor anterior o posterior al valor mostrado. El jugador tiene tres opciones a elegir, y solo es correcta una de ellas. Para ajustar la dificultad, dos de las soluciones que se proponen siempre varían en más o menos uno del valor del enunciado, la tercera opción propuesta varía en más o menos diez. De esta forma se consigue ajustar más las opciones, forzando al jugador a pensar la respuesta.



Ilustración 9: escenario del juego anterior o posterior

## Unidades, decenas y centenas

El objetivo de este juego es encontrar las naranjas para cargarlas en la camioneta. El jugador debe indicarle la posición al frutero donde se encuentra la naranja, si en las unidades, en las decenas o en las centenas. Inicialmente el juego empieza con tres frutas, pero según se va acertando la dificultad aumenta incrementado en número de frutas hasta alcanzar un máximo de seis. Cada 20 aciertos, se añade una fruta más. En consecuencia, también se incrementan el número de soluciones posibles, añadiendo las unidades de millar, las decenas de millar y las centenas de millar. Además, las opciones de respuesta ofrecidas desde el inicio de la partida varían su posición, es decir, no se muestran en orden, ya que sería muy fácil si mantuvieran el orden real de centena, decenas y unidades.

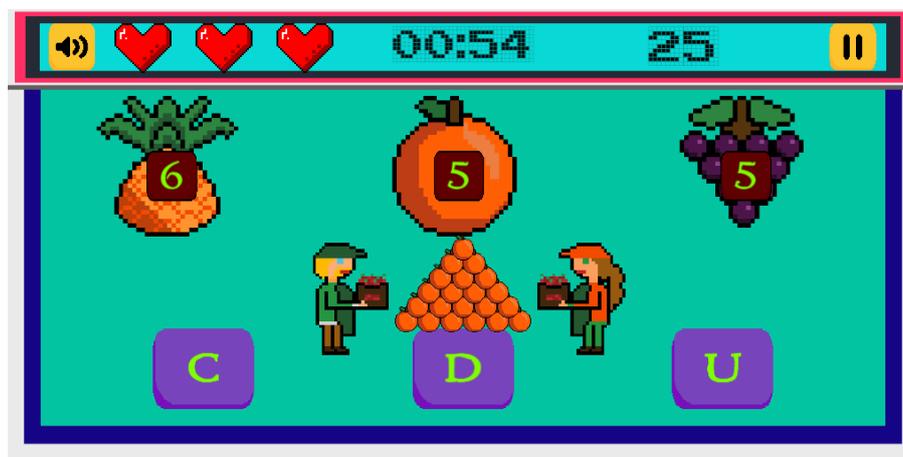


Ilustración 10: escenario del juego unidades, decenas o centenas

## Números romanos

El objetivo del juego de número romanos es ayudar al matemático a descifrar la cantidad de fruta que representa el número romano. El jugador tiene tres opciones, pero sola una de ellas representa el valor arábigo correspondiente al número romano. La dificultad viene establecida por el rango de valores sobre los que oscila el número romano mostrado, inicialmente este rango puede ser un valor entre 1 y 1000. Al alcanzar los 20 aciertos, el rango llega hasta 3999, este valor es el máximo que se puede alcanzar con los siete símbolos que ofrece la numeración romana (mayor número romano representado en el juego, MMMCMXCIX).

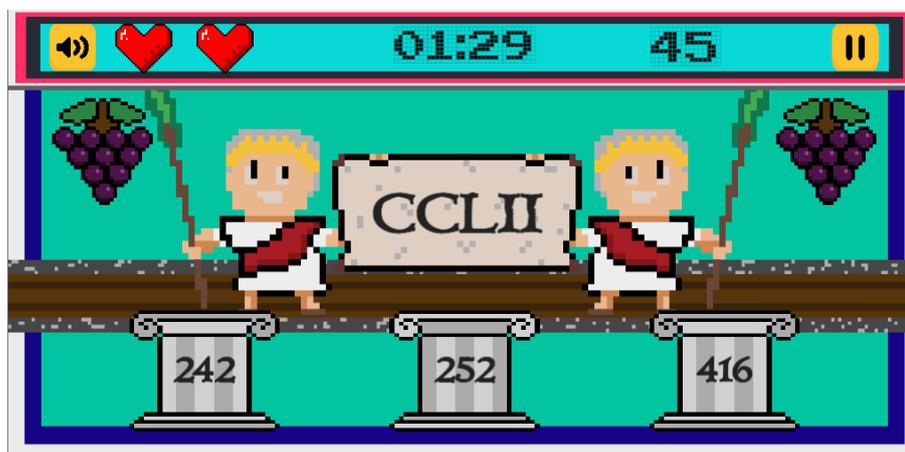


Ilustración 11: escenario del juego números romanos

## Esquema de control

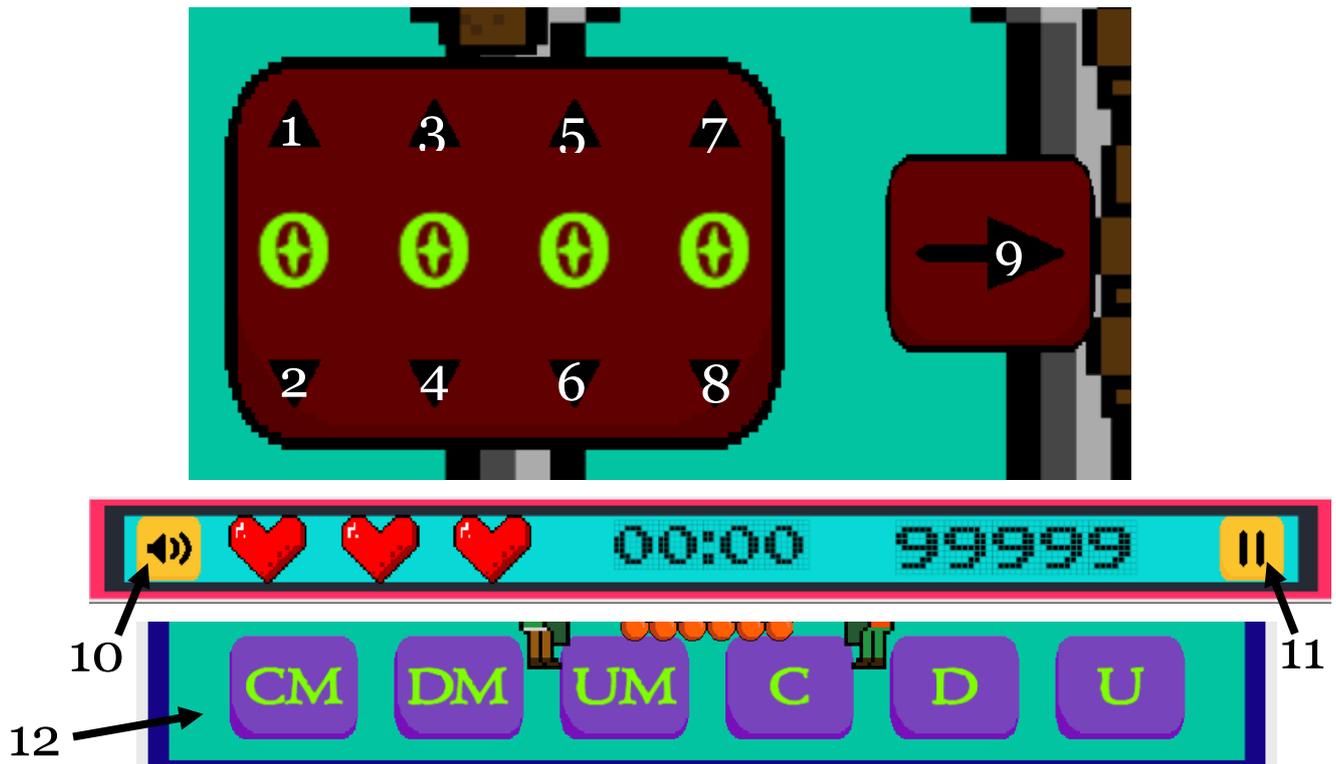


Ilustración 12: controles

Botón	Acción que realiza
1	Subir una unidad de millar
2	Bajar una unidad de millar
3	Subir una centena
4	Bajar una centena
5	Subir una decena
6	Bajar una decena
7	Subir una unidad
8	Bajar una unidad
9	Comprobar solución
10	Silenciar
11	Abrir/Cerrar menú de pausa
12	Respuesta elegida como resultado del juego

## Estética del juego e interfaz de usuario

El juego tiene una estética infantil con muchos colores (utilización de distintas frutas como decoración de la escena) y diseño pixelado como los antiguos juegos arcade. Dominan los colores azules y verdes para los fondos ya que es un juego educativo para niños y estos colores ayudan a mejorar la concentración, la comprensión lectora a la vez que aportan vitalidad y energía.

### Menú principal

En el menú principal del juego, nos encontramos arriba a la izquierda al botón para silenciar los todos los sonidos, en la parte central, el botón de jugar para empezar a elegir el juego al que queremos jugar, el botón para acceder al menú de trofeos y consultar todos los trofeos conseguidos y por conseguir, un botón de ajustes para ajustar la música de fondo y efectos, y por último un botón para salir del videojuego.



Ilustración 13: menú principal

## Menú de contenidos

Al pulsar el botón jugar del menú principal accedemos al menú de contenidos. Aquí es donde se dividen los dos grandes bloques matemáticos en los cuales se divide el videojuego, el bloque de numeración y el bloque de cálculo mental y operaciones. Según al bloque que accedamos disponemos de juegos diferentes.

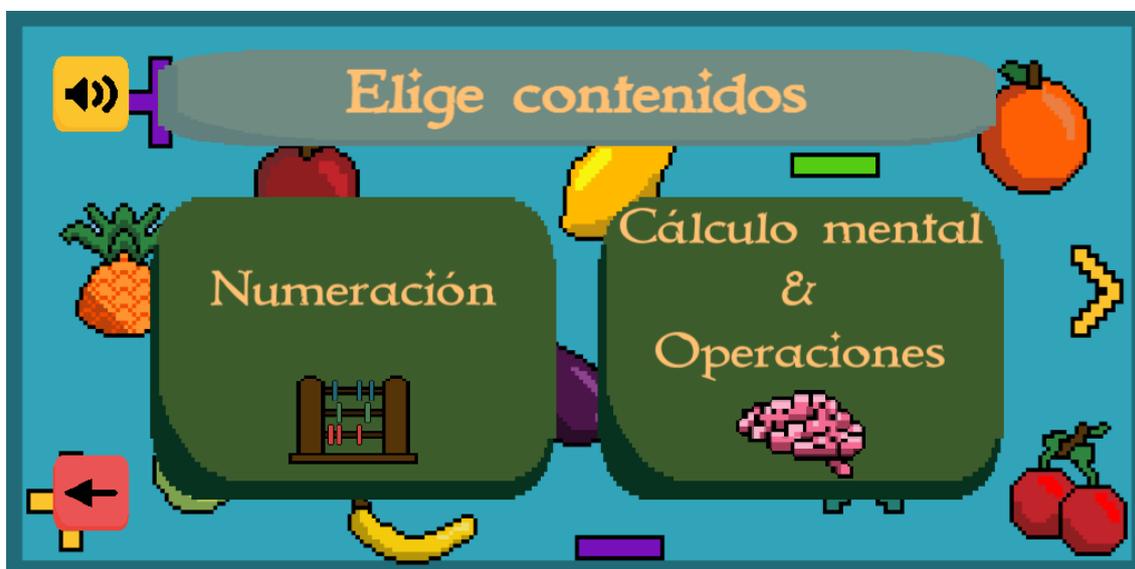


Ilustración 14: menú contenidos

## Menú numeración

Es el menú desde el cual se puede acceder a los seis juegos de numeración disponibles, el juego de mayor o menor, el fracciones, par o impar, anterior o posterior, unidades, decenas y centenas y por último números romanos.

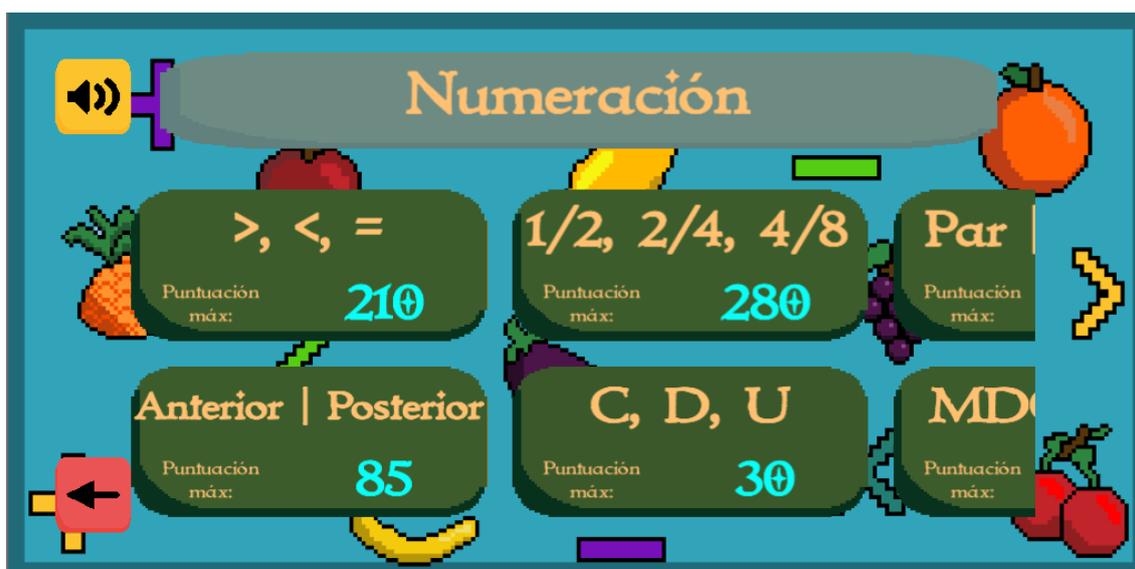


Ilustración 15: menú numeración

## Menú de cálculo mental & operaciones

Es el menú desde el cual se puede acceder a los cuatro juegos de cálculo mental y operaciones disponibles, el juego de suma, resta, multiplicación y división.



Ilustración 16: menú cálculo mental y operaciones

## Menú de instrucciones

Al seleccionar un juego nos encontramos con el menú de instrucciones donde se muestra el objetivo del juego y un mensaje de recuerda que puede ser un consejo a la hora de jugar o una propiedad matemática. Para cada juego, el menú de instrucciones muestra tanto el objetivo como el recuerda asociado a ese juego.



Ilustración 17: menú instrucciones

## HUD

El HUD es igual para los 10 modos de juego distintos. Empezando de izquierda a derecha: lo primero que nos encontramos es el botón de silenciar, a continuación las 3 vidas que disponemos en todas las partidas, en el centro tenemos el tiempo restante de juego antes de que se termine y perdamos, a su derecha tenemos la puntuación y por último a la izquierda tenemos el botón de pausa.

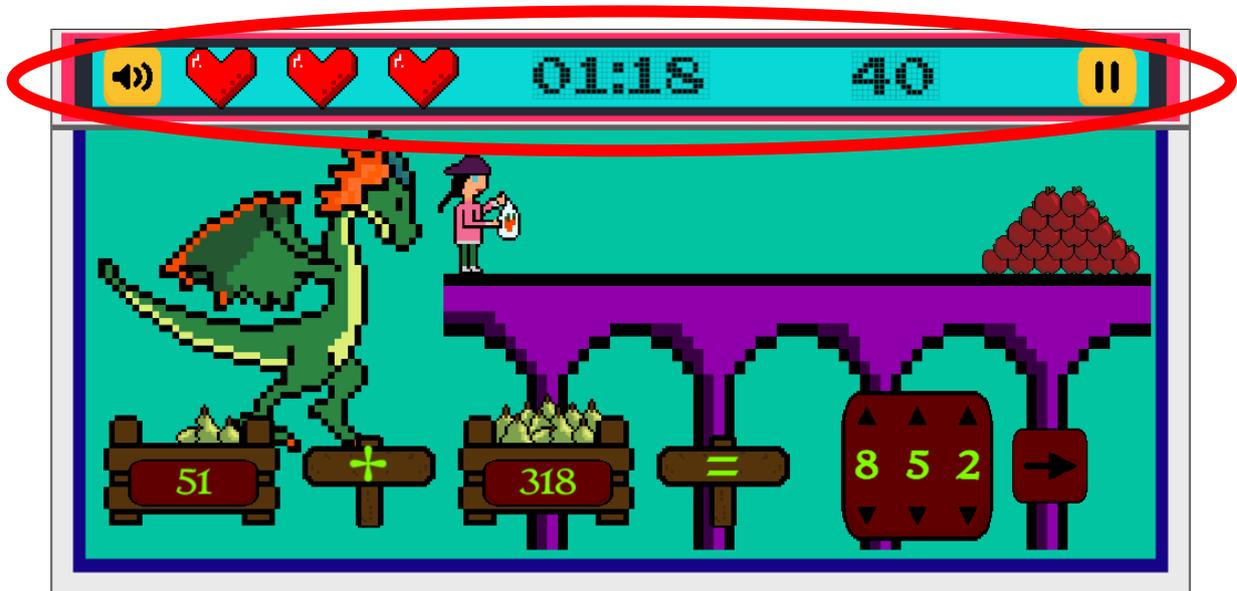


Ilustración 18: HUD del videojuego

## Menú de pausa

Al pulsar el botón de pausa se abre el menú de pausa desde el cual se puede consultar el objetivo del juego, continuar la partida, reiniciar la partida desde cero o salir de la partida al menú de contenidos. Al pausar la partida, el HUD continúa visible para el jugador pero el tiempo permanece pausado hasta continuar la partida.

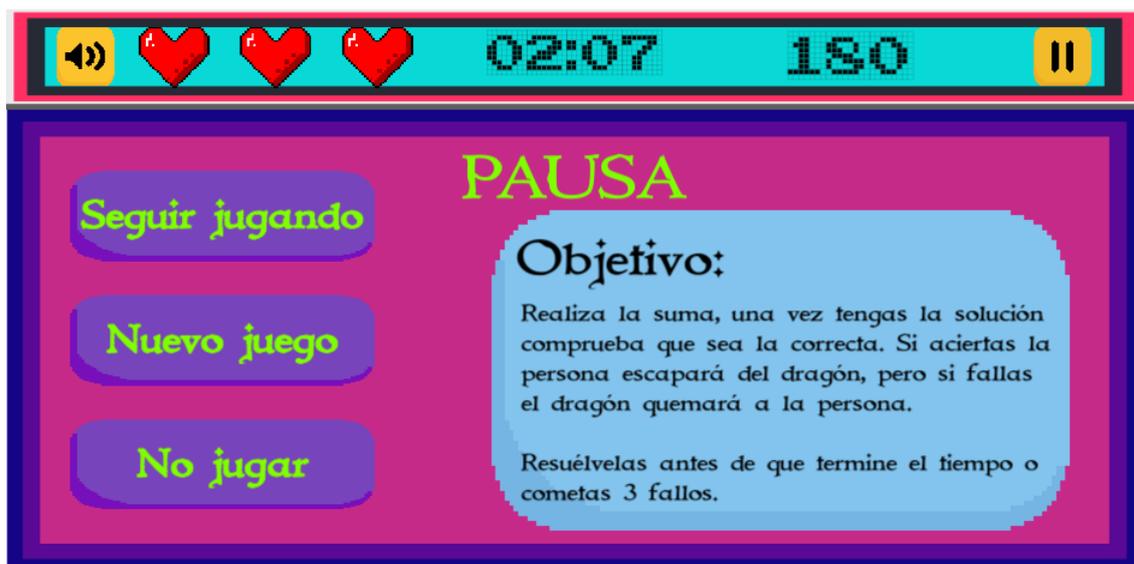


Ilustración 19: menú de pausa

### Menú de finalización de partida

Al finalizar una partida, ya sea por haber perdido las tres vidas, agotarse el tiempo o salir, se muestran las estadísticas de la partida que pueden interesar al jugador, tales como: la puntuación total obtenida, el tiempo total de juego y los aciertos conseguidos. Tanto la puntuación como los aciertos son la clave para conseguir trofeos.

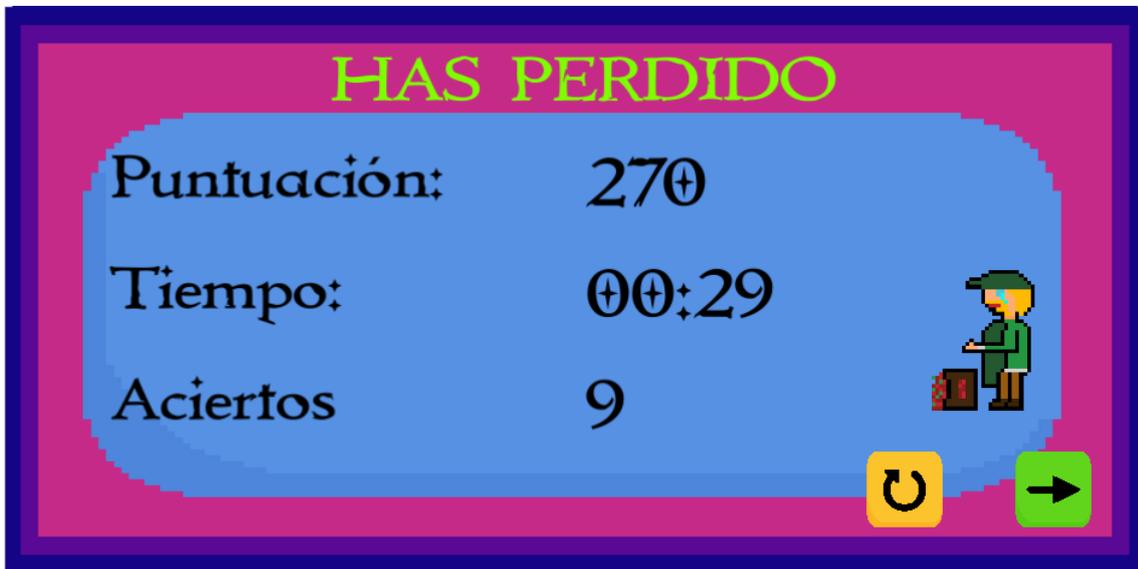


Ilustración 20: menú finalización de partida

### Menú de trofeos

En el menú de trofeos se pueden consultar todos los trofeos conseguidos y por conseguir.

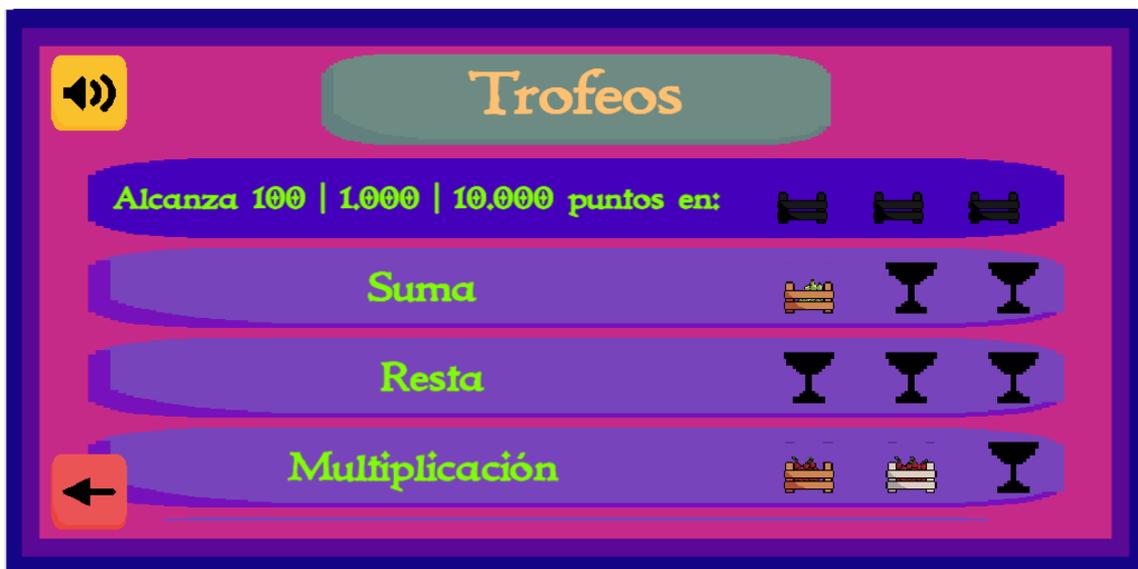


Ilustración 21: menú de trofeos

## Menú de ajustes

Desde el menú de ajustes se puede configurar tanto el volumen de la música de fondo como de los efectos sonoros del videojuego.



Ilustración 22: menú de ajustes

## Diagrama de navegación

A continuación, podemos ver el diagrama de navegación, el cual nos muestra cómo podemos navegar desde las distintas pantallas.

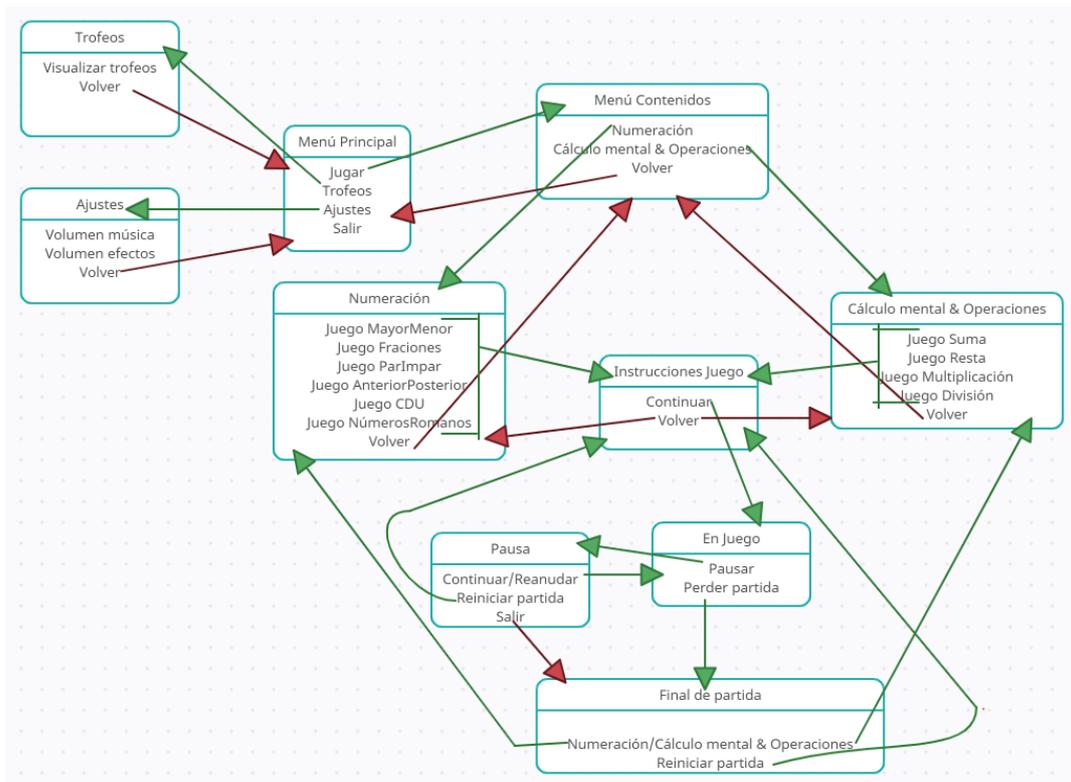


Ilustración 23: diagrama de navegación

## Personajes del juego

Personajes principales: el frutero y la frutera que recibirán la ayuda del jugador y estarán presentes en todos los juegos excepto el de números romanos.



Ilustración 24: frutero y frutera

Personajes a salvar junto con los dos de arriba en el juego de suma y resta.

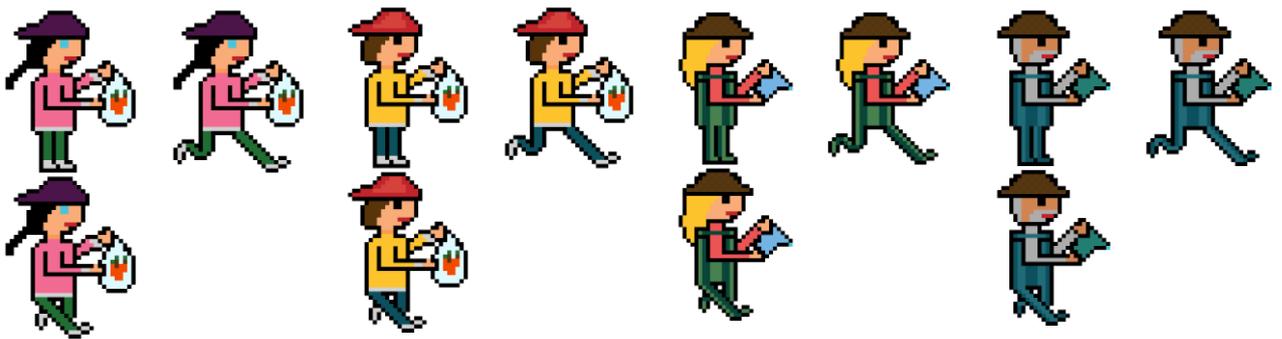


Ilustración 25: personajes a salvar



El dragón enemigo en los juegos de suma y resta.

Ilustración 26: dragón

Matemáticos y soldados del juego de números romanos

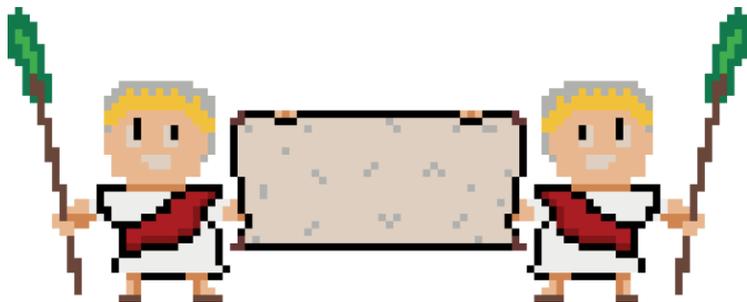


Ilustración 27: matemáticos romanos



Ilustración 28: soldado romano

Fruteros juego anterior o posterior

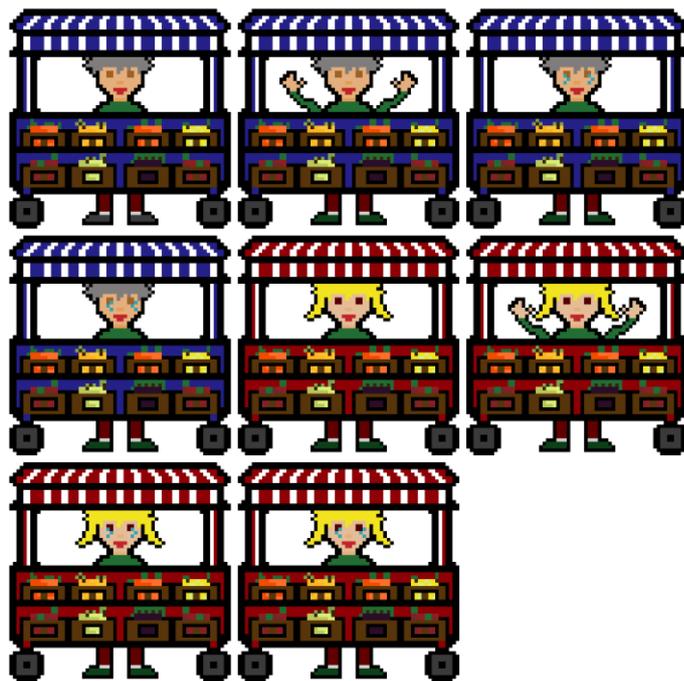


Ilustración 30: tienda frutero y frutera

Camioneta del reparto de fruta



Ilustración 29: camioneta fruta

## Programación y tareas

En el siguiente diagrama de Gantt pueden observarse todas las tareas llevadas a cabo y la duración de estas. Aunque no está escrito explícitamente en el nombre las tareas, cada tarea incluye la creación de los diseños y animaciones requeridas. En la memoria del TFG en el punto 3.5 de planificación se pueden leer más detalles sobre cada tarea.



Ilustración 31: diagrama de Gantt

Surgieron algunos contratiempos durante el desarrollo, entre ellos, en el bloque de numeración, inicialmente los cuatro juegos eran de elección y ofrecían tres opciones donde una era la solución correcta. Se decidió cambiar este sistema parecido más a un examen tipo test, por el sistema explicado en el juego suma, donde el jugador debe hacer la operación e ir colocando uno a uno los valores del resultado, esto implica un mayor grado de atención y pensamiento por parte del jugador. Estos cambios se aplicaron en el mes de junio tras finalizar el juego, puesto que se vio mejor idea añadir este nuevo sistema.

Posteriormente a la creación del videojuego se invirtió tiempo en pruebas, tanto del juego en sí mismo, como en niños de entre 7 y 11 años para obtener resultados que validaran la eficacia del videojuego educativo. Tras estas pruebas, en el mes de julio se aplicaron cambios moderados sobre en el videojuego en el texto de algunos botones, la tipografía de la letra y el balance de la dificultad del modo multiplicación, pues la mayoría de los niños coincidían en que este modo era el más difícil.

## Derechos de autor

Algunos de los elementos utilizados en el videojuego pueden estar sujetos a derechos de autor. Entre ellos las fuentes de texto utilizadas son fuentes gratuitas de la página *1001FreeFonts*. La música utilizada en el juego también se debe agradecer a Patrick Arteaga el cual tiene una página web (*patrickdearteaga.com*) donde ofrece gran variedad de música de fondo de forma gratuita. Los sonidos utilizados para los efectos del videojuego se han obtenido de *Sonidosmp3gratis*.

