

Treball final de grau

Estudi: Grau en Disseny i Desenvolupament de Videojocs

Títol: Kingdom

Document: TFG

Alumne: Anabel Jesús Lorenzo

Tutor: Dra. Boada Oliveras, Immaculada

Departament: INFORMÀTICA, MATEMÀTICA APLICADA I ESTADÍSTICA

Àrea: LLENGUATGES I SISTEMES INFORMÀTICS

Convocatoria: 2022/09

Contents

| | | |
|----------|--|-----------|
| 1 | Introducción, motivaciones, proposito y objetivos del proyecto y distribución de tareas | 4 |
| 1.1 | Introducción | 4 |
| 1.2 | Motivaciones | 4 |
| 1.3 | Proposito y objetivos del proyecto | 4 |
| 1.4 | Distribución de tareas | 5 |
| 2 | Estudio de viabilidad | 5 |
| 2.1 | Viabilidad técnica, económica y legal | 5 |
| 2.2 | Perfil del jugador | 6 |
| 2.3 | Estado del arte | 6 |
| 2.3.1 | Analisis | 6 |
| 2.3.2 | Videojuegos más representativos | 8 |
| 3 | Planificación | 11 |
| 3.1 | Planificación Inicial | 11 |
| 3.2 | Organigrama | 13 |
| 3.3 | Planificación | 14 |
| 3.4 | Metodología de trabajo | 16 |
| 4 | Marco de trabajo y conceptos previos | 17 |
| 4.1 | Terminología | 17 |
| 4.2 | Pedagogía | 18 |
| 5 | Diseño del videojuego | 19 |
| 5.1 | Narrativa | 19 |
| 5.2 | Estudio y diseño de personajes | 20 |
| 5.2.1 | El Guía | 20 |
| 5.3 | Estética | 21 |

| | | |
|----------|--|-----------|
| 5.4 | Tecnología | 24 |
| 5.5 | Mecánicas | 24 |
| 5.5.1 | Espacio de juego | 25 |
| 5.5.2 | Jerarquía de retos | 26 |
| 5.5.3 | Objetos, recursos, atributos y estados | 28 |
| 5.5.4 | Economía del juego | 34 |
| 5.6 | Interfaces | 35 |
| 5.7 | Game layout charts | 42 |
| 5.7.1 | Estrategia de diseño a seguir | 42 |
| 5.7.2 | Diagramas de módulos | 42 |
| 6 | Implementación | 52 |
| 6.1 | Base de datos | 52 |
| 6.2 | Menús | 59 |
| 6.3 | Inputs | 62 |
| 6.4 | Minijuegos | 64 |
| 6.4.1 | Minijuego Dibujar el tono | 65 |
| 6.4.2 | Minijuego Gacha | 66 |
| 6.4.3 | Minijuego Escucha | 66 |
| 6.4.4 | Minijuego Memoria | 67 |
| 6.4.5 | Minijuego Ordenar Pinyin | 68 |
| 7 | Resultados | 69 |
| 7.1 | Minijuego Dibujar Tono | 69 |
| 7.2 | Minijuego Gacha | 70 |
| 7.3 | Minijuego Escucha | 71 |
| 7.4 | Minijuego Memoria | 72 |
| 7.5 | Minijuego Ordenar Pinyin | 73 |
| 8 | Conclusiones | 74 |

9 Trabajo futuro

74

10 Bibliografía

75

1 Introducción, motivaciones, proposito y objetivos del proyecto y distribución de tareas

1.1 Introducción

El desarrollo de este proyecto se basará en la creación desde cero de un videojuego educativo para teléfonos móviles mediante el cual se enseñen conceptos básicos del idioma chino.

En la actualidad aprender idiomas abre muchas puertas; te puede permitir optar a puestos de trabajo donde el idioma principal no sea tu idioma nativo; te puede permitir viajar sumergiendo en la cultura; conocer a gente de todas partes; aprender historia...

En concreto, el chino es un idioma con mucho futuro ya que en la actualidad 1,118 millones de personas hablan este idioma, siendo el segundo más hablado del mundo después del inglés.

Este videojuego podría calificarse como juego serio, es decir que su proposito principal es diferente al de la mera diversión. En este caso su proposito es aprender chino mientras te diviertes.

Sus principales objetivos en lo que respecta al aprendizaje son:

- Aprender la transcripción fonética del mandarín.
- Comprender el orden de los trazos en la escritura china.
- Asentar los conocimientos partiendo de una base solida através de la práctica.

1.2 Motivaciones

Mi principal motivación para escoger realizar este tipo de proyecto han sido la propia experiencia a la hora de buscar un producto similar y no encontrar ninguno. Ya que no conseguí encontrar algún videojuego o aplicación con características similares a lo que yo estaba buscando.

Así pues, me propuse a mi misma la creación de este videojuego uniendo mis dos pasiones, el idioma chino y los videojuegos.

1.3 Proposito y objetivos del proyecto

El principal proposito de la creación de este videojuego es la enseñanza de un idioma con más de cinco mil años de antigüedad del cual se han empapado todas las culturas circundantes a su país de origen: China.

En el caso de este proyecto, el videojuego se realizará para móviles y con objetivo pedagógico poniendo en práctica así los conocimientos aprendidos en las diversas asignaturas. Tanto el diseño del videojuego, el apartado artístico como la programación del videojuego serán realizados partiendo de los conocimientos previos obtenidos y uniendolos de manera coherente.

Así pues el producto final que se obtendrá, será una demo del videojuego que se explicará a lo largo de este documento.

1.4 Distribución de tareas

La distribución de tareas es equitativa en todos los ámbitos ya son todos necesarios para la creación de cualquier videojuego. En este caso concreto, para la demo le he dado igual importancia a todas las partes sin centrarme demasiado en un apartado abandonando el resto.

| | |
|------------|-----|
| Estética | 25% |
| Narrativa | 25% |
| Mecánicas | 25% |
| Tecnología | 25% |

2 Estudio de viabilidad

2.1 Viabilidad técnica, económica y legal

Para poder realizar este proyecto parto de un presupuesto base de 0€, así que las tecnologías que opté a utilizar son gratuitas o, con versión de prueba o, programas que ya poseía anteriormente.

En el ámbito de programación, utilicé la plataforma Unity como motor de videojuegos acompañada del Visual Studio Code como editor. Para la documentación, opté por utilizar Overleaf y Microsoft Power Point (versión de estudiantes). Además, en el ámbito artístico utilicé InkScape y Paint Tool Sai. Finalmente para llevar un control más exhaustivo de las tareas realizadas, utilicé la web Hack n Plan.

A parte del software gratuito, previamente disponía de un ordenador con el que poder realizar el videojuego, por lo tanto, la inversión en hardware, al igual que la inversión en software obtuvo un coste de 0€. Al ser un proyecto íntegramente desarrollado por mi, no necesité de contrataciones externas.

Por otra parte, los recursos utilizados son de libre uso o con licencias gratuitas o no comerciales, es por eso que legalmente se cumplen todas las condiciones necesarias para poder realizar el videojuego sin ningún problema. Además el videojuego incluirá la pegatina de PEGI 7, ya que cualquier persona a partir de esa edad debería ser capaz de jugar al juego sin ningún tipo de problema.

Viendo así que los apartados técnico, económico y legal están cubiertos, podemos decir que un producto viable para el desarrollo.

2.2 Perfil del jugador

Este videojuego está orientado a jóvenes, de entre 16 y 29 años, interesados en el idioma Chino y que quieran aprender este de una manera diferente y entretenida. Así pues, como perfil de jugador no nos centramos en personas que ya hayan jugado antes a videojuegos ya que las mecánicas de los minijuegos son simples y fáciles de aprender para todo tipo de personas.

Aun así, no se descarta la posibilidad que personas mayores de 30 años o menores de 15 años puedan utilizar este videojuego para aprender el idioma ya que, dada la sencillez de las interfaces y mecánicas el juego se vuelve accesible para personas de todas las edades siempre que estas tengan interés en aprender chino.

El jugador contará con una motivación intrínseca para aprender el idioma pues lo hará mientras se divierte y explora el mundo de juego. De esta manera, su aprendizaje se verá reflejado y recompensado dentro del juego de modo que, a medida que avance los niveles mejorará sus habilidades en el idioma chino.

2.3 Estado del arte

He realizado un análisis del estado del arte actual del mundo del aprendizaje del idioma chino. Para ello he investigado sobre diversos libros, artículos, videos, cursos online, juegos y otras aplicaciones serias y he obtenido los siguientes resultados.

2.3.1 Analisis

- Libros y Artículos

- Palabras clave *aprender chino*

Lugar de búsqueda: google.com

Resultados de la búsqueda: aproximadamente 48.300.000 resultados

- * Aprende chino online gratis. Lecciones con audio y pinyin

- * APRENDER CHINO MANDARÍN ES FÁCIL. TU SOLO PIENSAS QUE ES DIFÍCIL

- * Curso de Chino gratis en línea

Títulos más representativos:

- * ¿Cómo aprender chino por nosotros mismos?

- Palabras clave *learn chinese*

Lugar de búsqueda: google.com

Resultados de la búsqueda: aproximadamente 2.560.000 resultados

- * Learning gap: Why our schools are failing and what we can learn from Japanese and Chinese educ

- * Measuring motivation to learn Chinese and English through self-reported feelings and behaviours

- * A cultural model of learning: Chinese "hear and mind for wanting to learn"

Títulos más representativos:

- * Associations between Chinese Language Classroom Environments and Students' Motivation to Learn the Language

- Videos

- Palabras clave *Learn chinese radicals*

Lugar de búsqueda: [google.es](https://www.google.es)

Resultados de la búsqueda: aproximadamente 192.000 resultados

- * The FASTEST Way to Learn Chinese Radicals
- * Storied Mandarin: The Creative Guide to Chinese Radicals
- * The Chinese Radicals — HSK Academy

Títulos más representativos:

- * All Chinese Radicals With Stroke Order Animations

- Palabras clave *Aprender radicales chinos*

Lugar de búsqueda: 117.000

Resultados de la búsqueda: aproximadamente N^o resultados

- * Conocer Los Radicales Escenciales del Chino — Aprender chino, Escritura China, Curso de chino
- * basicos de la escritura china - practicas los trazos y radicales
- * TODO LO QUE TIENES QUE SABER DE LOS CARACTERES CHINOS

Títulos más representativos:

- * ¿Qué es un radical? - Clasificador de hanzi y mucho más

- Cursos-online

- Palabras clave *curso chino online*

Lugar de búsqueda: [google.es](https://www.google.es)

Resultados de la búsqueda: aproximadamente 10.100.000 resultados

- * Curso de Chino Mandarín Online — Hanyu Chinese School
- * Curso de Chino Online - Aprende Chino con UOC X - UOC X
- * Curso online Chino General . Cambridge Institute

Títulos más representativos:

- * Aprender Chino online — Draggon - Escuela de chino online

- Palabras clave *chinese course online*

Lugar de búsqueda: [google.es](https://www.google.es)

Resultados de la búsqueda: aproximadamente 813.000.000 resultados

- * Chinese for Beginners — Coursera
- * Beast (and Worst) Online Chinese Courses - From Personal Experience
- * Lean Chinese with Online Course and Lessons— edX

Títulos más representativos:

- * The Most Recommended Online Chinese Couses

- Juegos y aplicaciones serias

- Palabras clave *chinese learning game*

Lugar de búsqueda: [google.es](https://www.google.es)

Resultados de la búsqueda: aproximadamente 307.000.000 resultados

- * Mandarin Games - PandaTree
- * Mandarin Chinese language learning games — Digital Dialects
- * Let the Games Begin! 9 Chinese Learning Games for All Levels of Learners

Títulos más representativos:

- * Games Learn Chinese
- Palabras clave *chinese learn game*
Lugar de búsqueda: <https://play.google.com/>
Resultados de la búsqueda: no ofrece un número de resultados
- * Fun Chinese: Aprende Chino
- * Aprende Chino AI-Super Chinese
- * Infinite Chinese
- * Learn Chinese for Kids
- * M Mandarin-漫中文- Learn Chinese - Aprender Chino HSK
- * Drops: ¡aprende chino mandarín gratis

Títulos más representativos:

- * Aprende chino - HelloChinese

2.3.2 Videojuegos más representativos

Análisis de los videojuegos de aprender chino más representativos y porque escogí estos.

- Fun Chinese: Aprende Chino. Incluye una gran variedad de minijuegos y es muy visual.
 - Pros:
 - * Tiene un gran número de minijuegos.
 - * Realizan muchas repeticiones de sonido.
 - Contras:
 - * No enseñan los trazos ni el hanzi.
 - * Está orientado a niños pequeños.
 - * El 90% de la aplicación es de pago

- Infinite Chinese. En una aplicación con un único minijuego muy entretenido.
 - Pros:
 - * Requiere velocidad y agilidad
 - * Separa el aprendizaje por categorías
 - Contras:
 - * Solo tiene un minijuego
 - * No incluye los hanzi, solo el pinyin

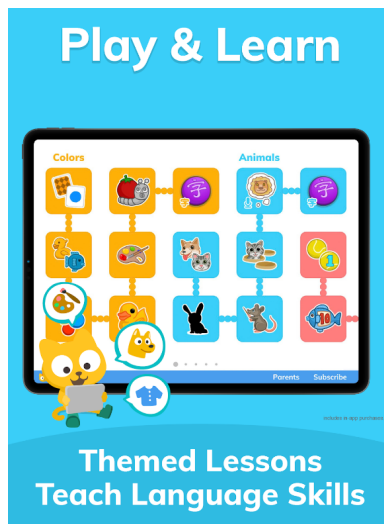


Figure 1: Fun Chinese



Figure 2: Infinite Chinese

- Aprende chino - HelloChinese. Es una aplicación que incluye una rotación de juegos (uno diferente cada día) que te permiten practicar pronunciación, vocabulario y gramática.
 - Pros:
 - * Incluye lecciones ordenadas y los minijuegos se basan en el contenido aprendido en las lecciones
 - * Incluye una gran variedad de minijuegos
 - Contras:
 - * No puedes seleccionar el minijuego que quieres jugar a no ser que pagues
 - * No puedes seleccionar qué vocabulario en concreto quieres practicar en los minijuegos



Figure 3: Hello Chinese

| | Fun Chinese | Infinite Chinese | HelloChinese | KINGDOM |
|-----------------|-------------|------------------|--------------|---------|
| Entretenimiento | 8 | 9 | 6 | 8 |
| Feedback | 8 | 7 | 8 | 9 |
| Aprendizaje | 6 | 3 | 10 | 10 |
| Práctica | 8 | 6 | 5 | 10 |
| Modos de juego | 10 | 1 | 6 | 10 |

En líneas generales, podemos observar que los minijuegos ofrecidos por las aplicaciones (si es que ofrecen) son muy simples, en muchas ocasiones no se centran en el orden de trazos del hanzi y, normalmente, no enseñan los radicales a diferencia de mi videojuego.

3 Planificación

La planificación del videojuego esta dividida en paquetes de trabajo los cuales incluyen: la documentación, la estética, la narrativa, las mecánicas, la tecnología y la integración y testo.

3.1 Planificación Inicial

Unidad de tiempo utilizada: días. 1 día equivale a 5 horas.

- Paquete de trabajo 1: Documentación
 - Equipo de trabajo: Anabel
 - Tareas:
 1. Introducción
 2. Estudio de viabilidad
 3. Planificación
 4. Marco de trabajo y conceptos previos
 5. Diseño del videojuego
 6. Conclusiones
 7. Trabajo futuro
 - Temporización: 01/01/2021 - 31/01/2021
 - Hito: Documento
- Paquete de trabajo 2: Estética
 - Equipo de trabajo: Anabel
 - Tareas:
 1. Diseño de personajes
 2. Diseño de interfaces
 3. Diseño de niveles
 - Temporización: 15/02/2021 - 15/05/2021
 - Hito: Assets
- Paquete de trabajo 3: Narrativa
 - Equipo de trabajo: Anabel
 - Tareas:
 1. Guión
 - Temporización: 06/02/2021 - 09/02/2021
 - Hito: Documento
- Paquete de trabajo 4: Mecánicas
 - Equipo de trabajo: Anabel

- Tareas:
 1. Pantalla inicio
 2. Menús y otros
 3. Mundo de juego
 4. Modo práctica
 5. Modo historia
 6. Minijuegos
- Temporización: 01/02/2021 - 15/05/2021
- Hito: Mecánicas implementadas
- Paquete de trabajo 3: Tecnología
 - Equipo de trabajo: Anabel
 - Tareas:
 1. Selección de entorno de trabajo
 - Temporización: 27/01/2021 - 31/01/2021
 - Hito: Entorno de trabajo seleccionado
- Paquete de trabajo 4: Integración y testeo
 - Equipo de trabajo: Anabel
 - Tareas:
 1. Integración
 2. Testeo
 - Temporización: 08/05/2021 - 06/06/2021
 - Hito: Juego completo

3.2 Organigrama

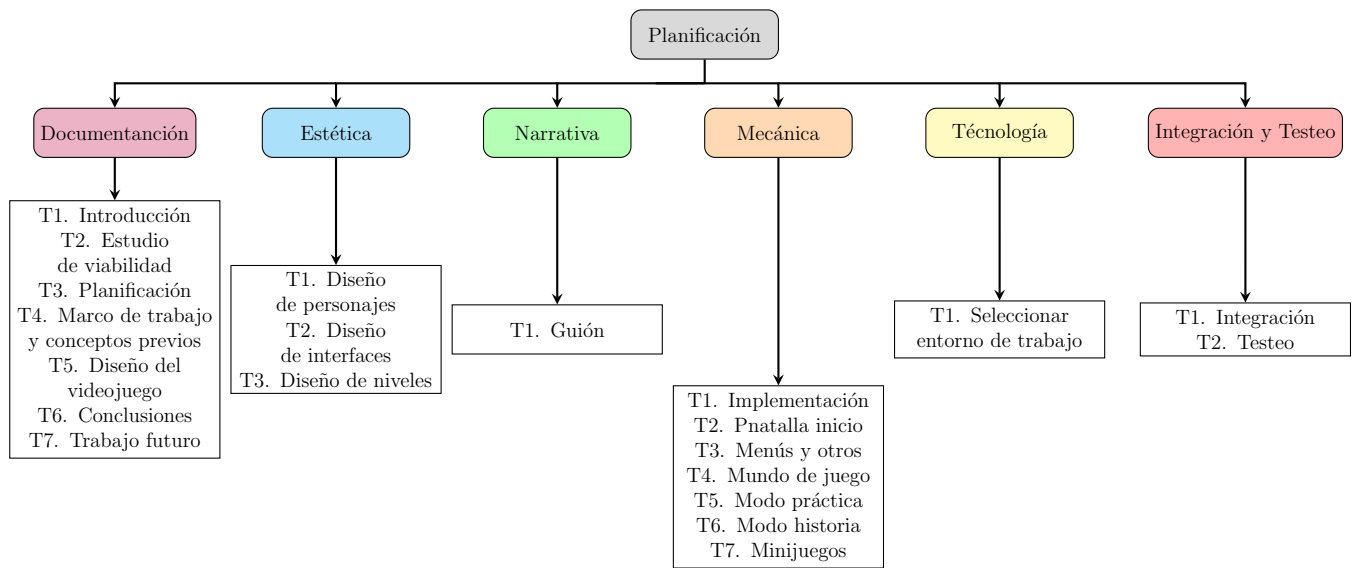


Figure 4: Organigrama

3.3 Planificación

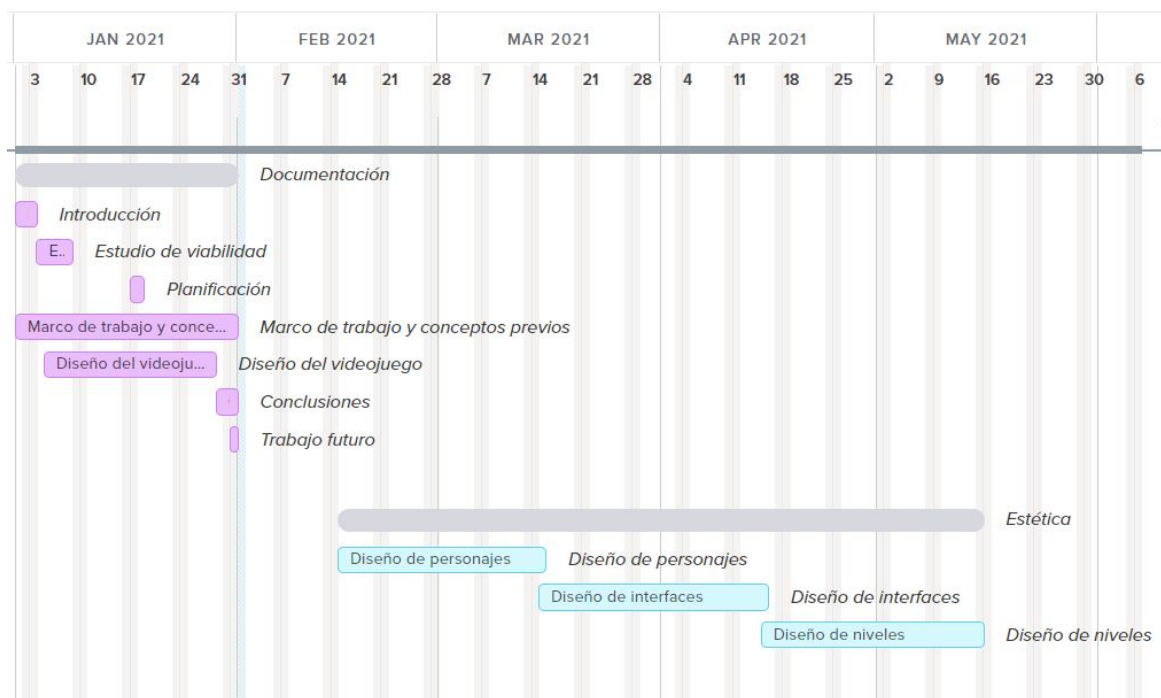


Figure 5: Planificación: Documentación y Estética

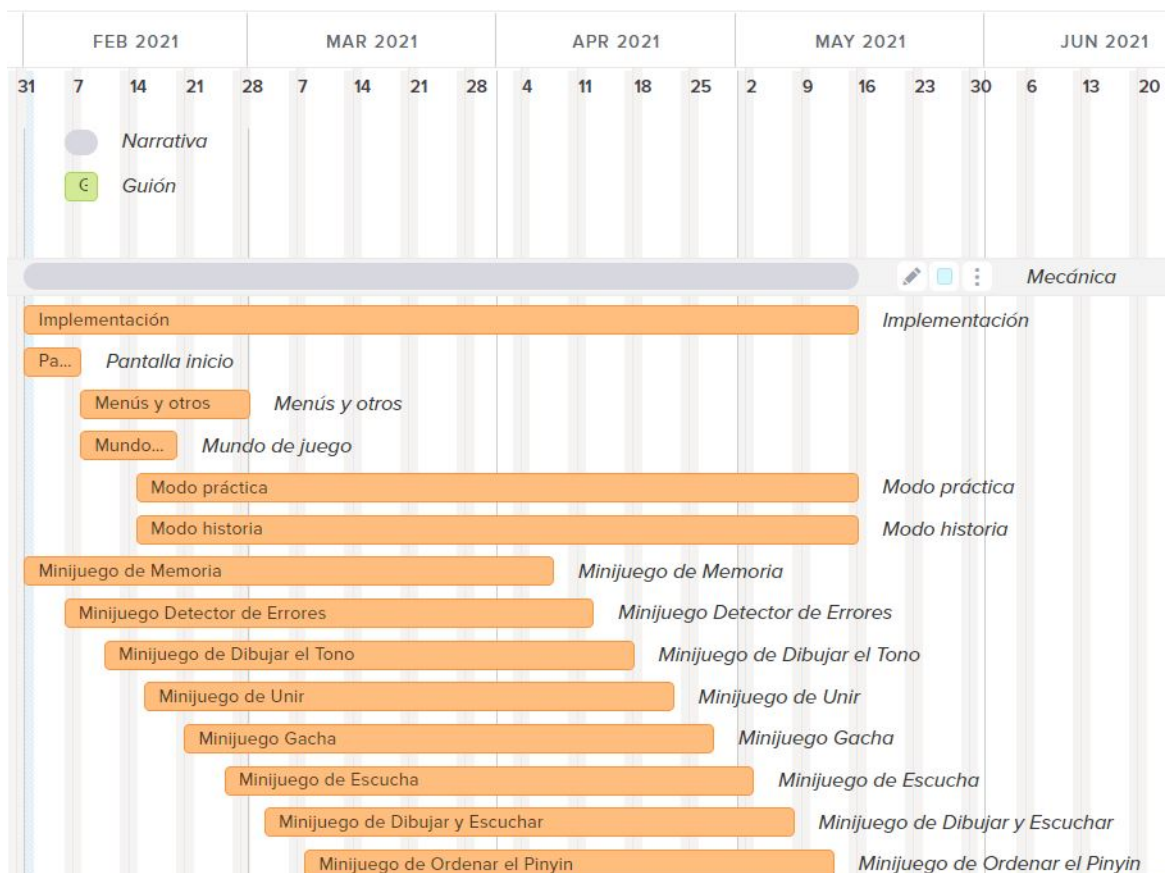


Figure 6: Planificación: Narrativa y Mecánica

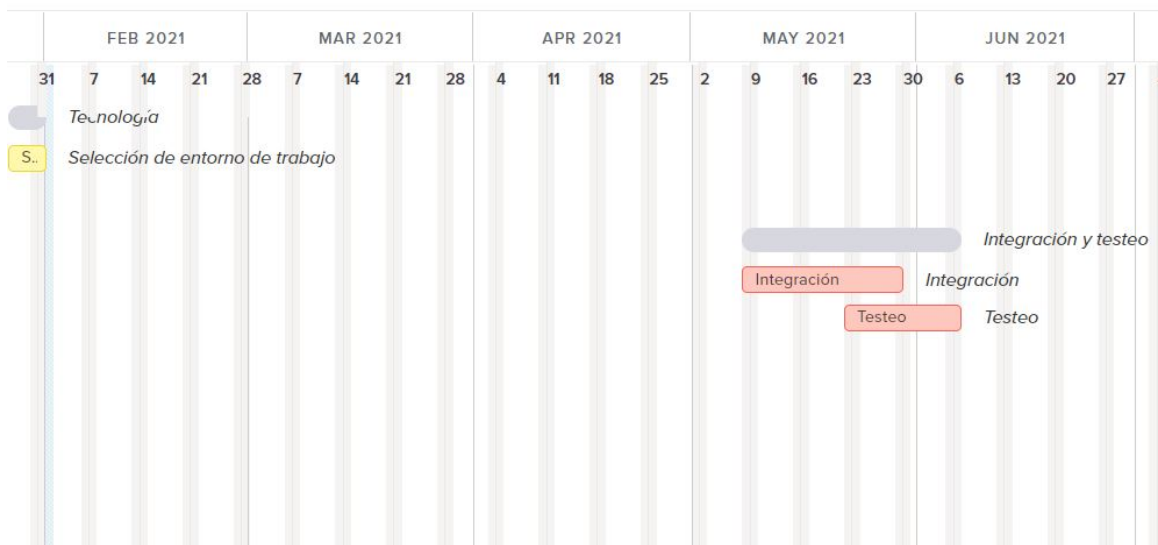


Figure 7: Planificación: Tecnología e Integración y Testeo

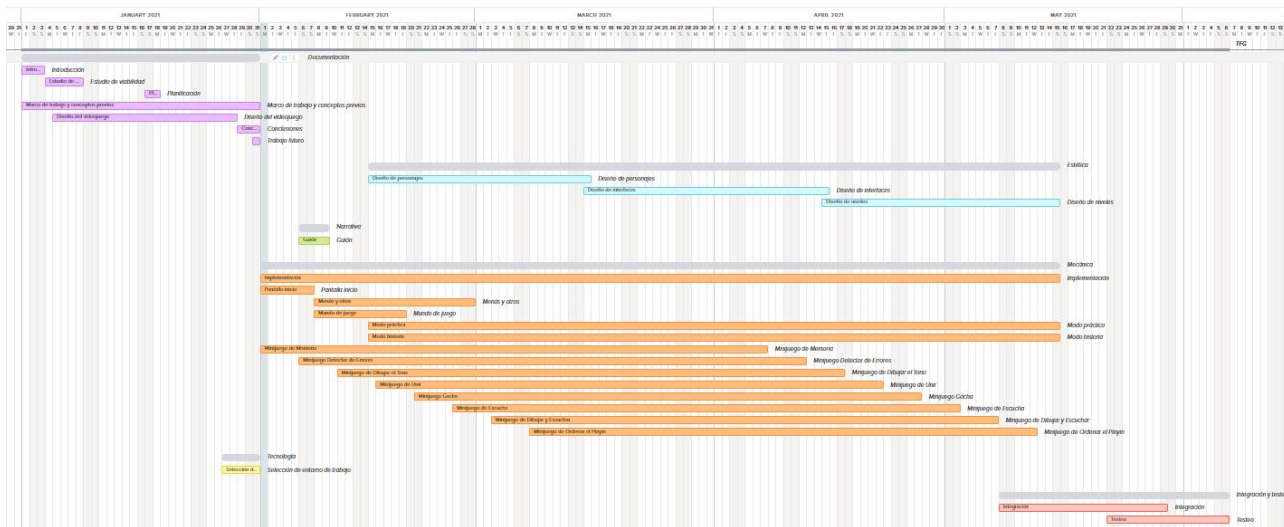


Figure 8: Planificación: Visión global

3.4 Metodología de trabajo

El documento de trabajo está íntegramente realizado por Anabel Jesús Lorenzo (al igual que el videojuego lo estará en el futuro).

Iniciaré el desarrollo del videojuego realizando de manera paralela la estética y la implementación de las mecánicas y al acabar esta parte, iniciaré el testeo para que, en caso de haber algún bug solucionarlo.

A medida que se van desarrollando las distintas tareas estas se han ido testeando e integrando en caso de ser necesario. Además, una vez al mes tendré una reunión con la tutora para actualizarla del estado del proyecto y una comunicación periodica por email cuando surjan dudas sobre el desarrollo o sobre el documento.

| TFG | |
|--------------------------------------|--------------------|
| ▼ Documentación | |
| Introducción | Anabel Jesús Loren |
| Estudio de viabilidad | Anabel Jesús Loren |
| Planificación | Anabel Jesús Loren |
| Marco de trabajo y conceptos previos | Anabel Jesús Loren |
| Diseño del videojuego | Anabel Jesús Loren |
| Conclusiones | Anabel Jesús Loren |
| Trabajo futuro | Anabel Jesús Loren |
| ▼ Estética | |
| Diseño de personajes | Anabel Jesús Loren |
| Diseño de interfaces | Anabel Jesús Loren |
| Diseño de niveles | Anabel Jesús Loren |
| ▼ Narrativa | |
| Guión | Anabel Jesús Loren |
| ▼ Mecánica | |
| Implementación | Anabel Jesús Loren |
| Pantalla inicio | Anabel Jesús Loren |
| Menús y otros | Anabel Jesús Loren |
| Mundo de juego | Anabel Jesús Loren |
| Modo práctica | Anabel Jesús Loren |
| Modo historia | Anabel Jesús Loren |
| Minijuego de Memoria | Anabel Jesús Loren |
| Minijuego Detector de Errores | Anabel Jesús Loren |
| Minijuego de Dibujar el Tono | Anabel Jesús Loren |
| Minijuego de Unir | Anabel Jesús Loren |
| Minijuego Gacha | Anabel Jesús Loren |
| Minijuego de Escucha | Anabel Jesús Loren |
| Minijuego de Dibujar y Escuchar | Anabel Jesús Loren |
| Minijuego de Ordenar el Pinyin | Anabel Jesús Loren |
| ▼ Tecnología | |
| Selección de entorno de trabajo | Anabel Jesús Loren |
| ▼ Integración y testeo | |
| Integración | Anabel Jesús Loren |
| Testeo | Anabel Jesús Loren |

Figure 9: Planificación: Visión global

4 Marco de trabajo y conceptos previos

4.1 Terminología

Durante la explicación del videojuego aparecerán palabras con las que el lector puede no estar familiarizado, es por eso que en este apartado se mostrará una breve explicación de los conceptos previos necesarios para comprender el documento.

- El término *pinyin* hace referencia al sistema de transcripción fonética del chino mandarín oficialmente reconocido en la República Popular China.
- El término *radical* hace referencia a cada uno de los 214 elementos en los que se pueden descomponer los caracteres chinos.
- El término *trazos* hace referencia a los puntos y líneas que componen los caracteres, es decir, cada vez que el bolígrafo o el pincel se apoya en el papel hasta que se levanta.
- El término *hanzi* hace referencia a los sinogramas, es decir, los caracteres chinos.
- El término *tono* hace referencia a la pronunciación de la sílaba en chino y existen cuatro tonos y el neutro.
- El término *e-learning* hace referencia a procesos de enseñanza-aprendizaje que se llevan a cabo a través de Internet, caracterizados por una separación física entre profesorado y estudiantes, pero con el predominio de una comunicación tanto síncrona como asíncrona, a través de la cual se lleva a cabo una interacción didáctica continuada. Además, el alumno pasa a ser el centro de la formación, al tener que autogestionar su aprendizaje, con ayuda de tutores y compañeros.
- El término *ítems* hace referencia a los objetos utilizables por el jugador.
- El término *HUD* (Head-Up Display o visualización cabeza-arriba, también conocido como barra de estado) es la información que en todo momento se muestra en pantalla durante la partida, generalmente en forma de iconos y números. El HUD suele mostrar el número de vidas, puntos, nivel de salud y armadura, minimapa, y otros, dependiendo del juego.
- El término *interfaz* hace referencia al medio con el que el usuario puede comunicarse con una máquina, equipo, computadora o dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo.
- El término *inventario* explica el espacio donde se encuentran los objetos del jugador.
- El término *assets* hace referencia a los archivos que se encuentran en el juego durante su desarrollo, por ejemplo los sprites.
- El término *sprite* hace referencia a las imágenes (usualmente con fondo transparente) utilizadas en el juego. Normalmente con una secuencia de estas imágenes se crea una animación.
- El término *feedback* es la capacidad de un emisor para recoger reacciones de los receptores y modificar su mensaje, de acuerdo con lo recogido.

- El término *sources* hace referencia a las mecánicas que permiten crear recursos nuevos de la nada.
- El término *drainers* es el término contrario a *sources*, eliminando recursos del juego, reduciendo la cantidad almacenada (de forma permanente).
- El término *converters* hace referencia a lo que convierte un recurso en otro (de otro tipo).
- El término *traders* hace referencia a las mecánicas que mueven un recurso de una entidad a otra a cambio de otro recurso.
- El término *testers* hace referencia a los jugadores que han probado la demo del juego.
- El término *pseudocodi* es una descripción de alto nivel compacta e informal del principio operativo de un programa informático u otro algoritmo.

4.2 Pedagogía

El objetivo pedagógico del juego se basa en:

- Objetivo 1: Aprender el sistema de transcripción fonética del mandarín: el pinyin. El jugador adquirirá los conocimientos a través de la historia del juego mediante minijuegos adaptados a la dificultad actual del mundo de juego.
- Objetivo 2: Aprender el orden de los trazos de la escritura china. El jugador adquirirá estos conocimientos a través de la historia del juego mediante minijuegos adaptados a la dificultad actual del mundo de juego.
- Objetivo 3: Conocer el pinyin y los trazos de los radicales que componen el idioma chino. El jugador adquirirá estos conocimientos a través de la historia del juego mediante minijuegos adaptados a la dificultad actual del mundo de juego.
- Objetivo 4: Consolidar los conocimientos del pinyin, el orden de trazos y los radicales. El jugador mejorará estos conocimientos a través de un modo práctica, al cual se le recomendará entrar cada día.

Los objetivos del juego son considerados SMART ya que son:

- Específicos: Aprender el orden de los trazos de los radicales, su principal significado, su pinyin y su pronunciación.
- Medibles: Son medibles en función de los fallos realizados y la velocidad de escritura.
- Logrables: Se pueden conseguir los objetivos mediante la práctica.
- Relevantes: Anteriormente (apartado 2.2) se demostró que es relevante.
- Acotados en el tiempo: no están acotados en el tiempo, pero, la falta de práctica puede hacer que el jugador olvide lo aprendido.

Así pues, utilizaremos el modelo pedagógico basado en las 5 metáforas del aprendizaje que son: la adquisición, el descubrimiento, la imitación y la experimentación.

Durante el proceso de aprendizaje del juego, aplicaremos diversos principios de Mayer para que el jugador obtenga los conocimientos de manera más clara y simple.

1. Principio de señalización Se le señalará al jugador qué debe hacer en caso de no saber como avanzar. (Mediante el símbolo (?)).
2. Principio de la redundancia Aparecerán de manera repetida los conceptos hasta que el jugador tenga los conceptos claros.
3. Principio de pre-entrenamiento Poseerá una fase de pre-entrenamiento en la que el jugador aprenderá el Pinyin y algunos conceptos básicos sobre el trazado de caracteres.
4. Principio de multimedia Se juntarán imágenes con textos para explicar mejor los conceptos.
5. Principio de la voz Mediante una lectura del texto que aparece escrito por una voz humana real.

5 Diseño del videojuego

Para presentar el diseño del videojuego consideraremos los cuatro bloques principales: la narrativa, la estética, las mecánicas y la tecnología.

5.1 Narrativa

El jugador llega nuevo a una ciudad de China y quiere realizar el examen imperial y para ello debe de aprender el chino de manera correcta. Durante su llegada, el jugador se encuentra con "El Guía" el cual le explica al jugador que los pergaminos de los hanzi se han perdido en el camino que sale del pueblo y por culpa de esto la biblioteca a la que el jugador quería acceder para poder estudiar esta vacía.

El jugador emprende su aventura por el camino que sale del pueblo obteniendo diversos pergaminos que representan los hanzi que se va aprendiendo. Pero el camino no es fácil y está lleno de pruebas (minijuegos) que el jugador debe de superar para poder obtener los pergaminos. Además de estos pergaminos, en algunas ocasiones el jugador se encuentra con bolsas repletas de tokens intercambiables por otros pergaminos.

Cuando el jugador obtenga la totalidad de los pergaminos, podrá al fin realizar el examen imperial el cual se basará en escribir una frase con los hanzi. Ya que con los pergaminos puede obtener más tokens que le permiten conseguir un mayor número de caracteres con los que escribir la frase.

Así pues, los escenarios disponibles para el jugador serán su actual casa donde almacena los pergaminos; una habitación de su casa donde almacena los tokens; la calle principal donde puede hablar con "El Guía"; la calle trasera donde el pueblo es más animado y está repleto

de tiendas donde el jugador puede practicar sus conocimientos; el camino al pueblo donde el jugador obtendrá los pergaminos; y finalmente, el castillo imperial donde se realizará el examen final.

5.2 Estudio y diseño de personajes

5.2.1 El Guía

- Peso narrativo del personaje

El peso narrativo de este personaje es de casi el 100% ya que es el personaje que guía al jugador por la historia.

- Los personajes como base de la jugabilidad

Respecto a la jugabilidad, este personaje ayudará y explicará al jugador como superar los diversos retos que se le planteen.

- Estilo artístico

Su apariencia es la de un erudito de China con las ropas tradicionales de estos.

- Coherencia, características físicas y psicología

Es un personaje muy sabio y con bastos conocimientos además de ser amable y generoso por lo que se ofrece a enseñarle sus conocimientos del mundo al jugador.

- Lenguaje corporal y gestual. Movimiento

Su lenguaje corporal y gestual es un tanto serio y reservado aunque a veces sonrío.

- Vestimenta

Las ropas tradicionales de los reuditos chinos varían en función de la dinastía de la que hablemos. Por ejemplo, en la dinastía Ming, las vestimentas eran:

- Objetos que caracterizan al personaje

Siempre lleva consigo un libro y una pluma (además de la tinta).



Figure 10: Erudito de la dinastía Ming

5.3 Estètica

La inspiración principal del videojuego son las pinturas y esculturas del mundo chino clasico. Donde se utilizan paletas de colores rojizas para mostrar elementos imperiales, de felicidad y fortuna, paletas de colores verdes para mostrar la calma y tranquilidad, además de representar el color del jade, una pieza muy valorada en china.

Algunos de los fondos que representan esta estética dentro del juego son los siguientes.

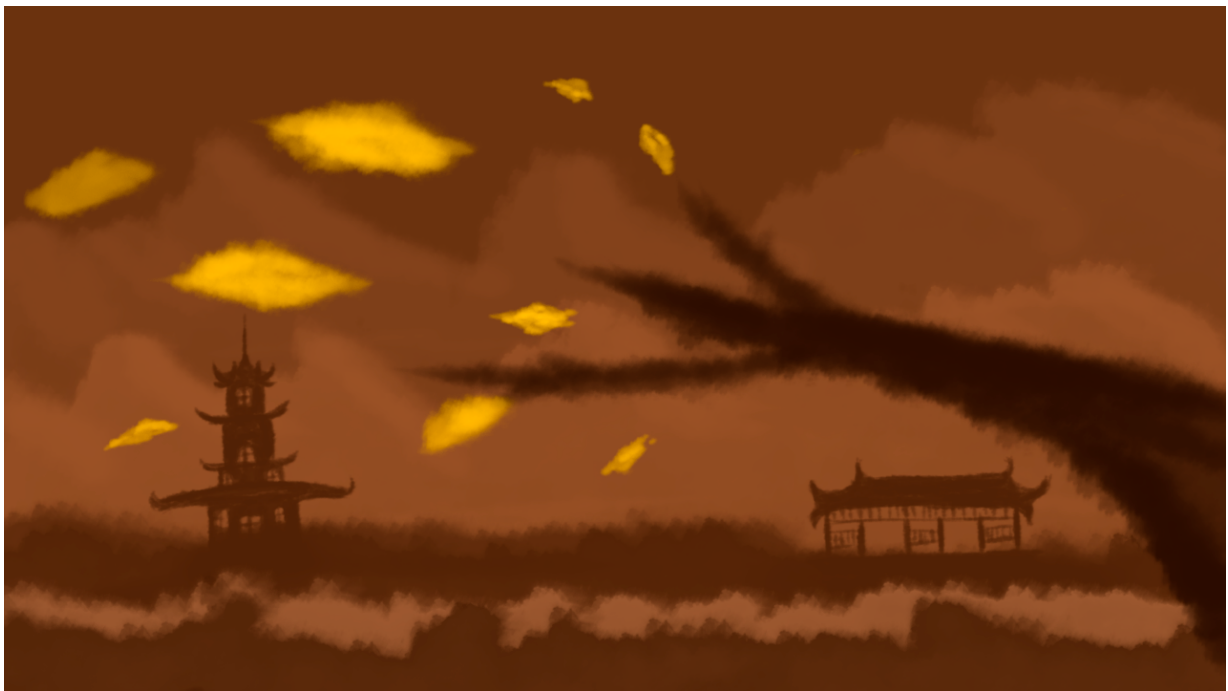


Figure 11: Fondo que aparece de forma aleatoria mientras juegas un minijuego

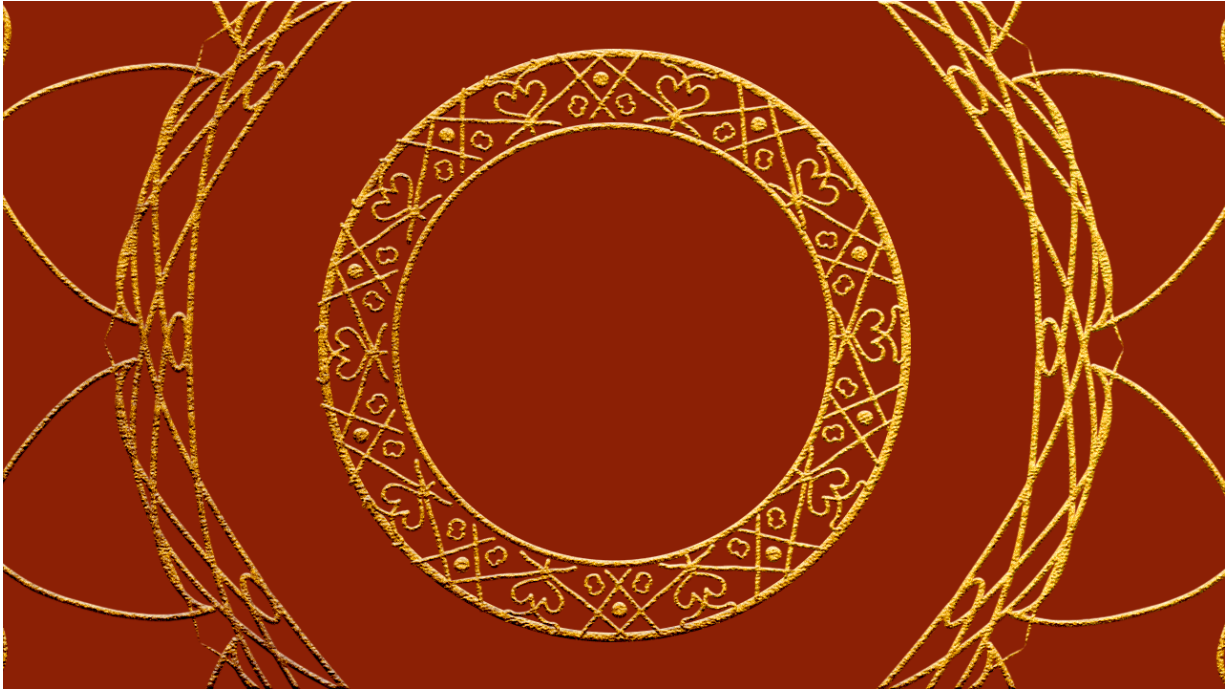


Figure 12: Fondo que aparece de forma aleatoria mientras juegas un minijuego

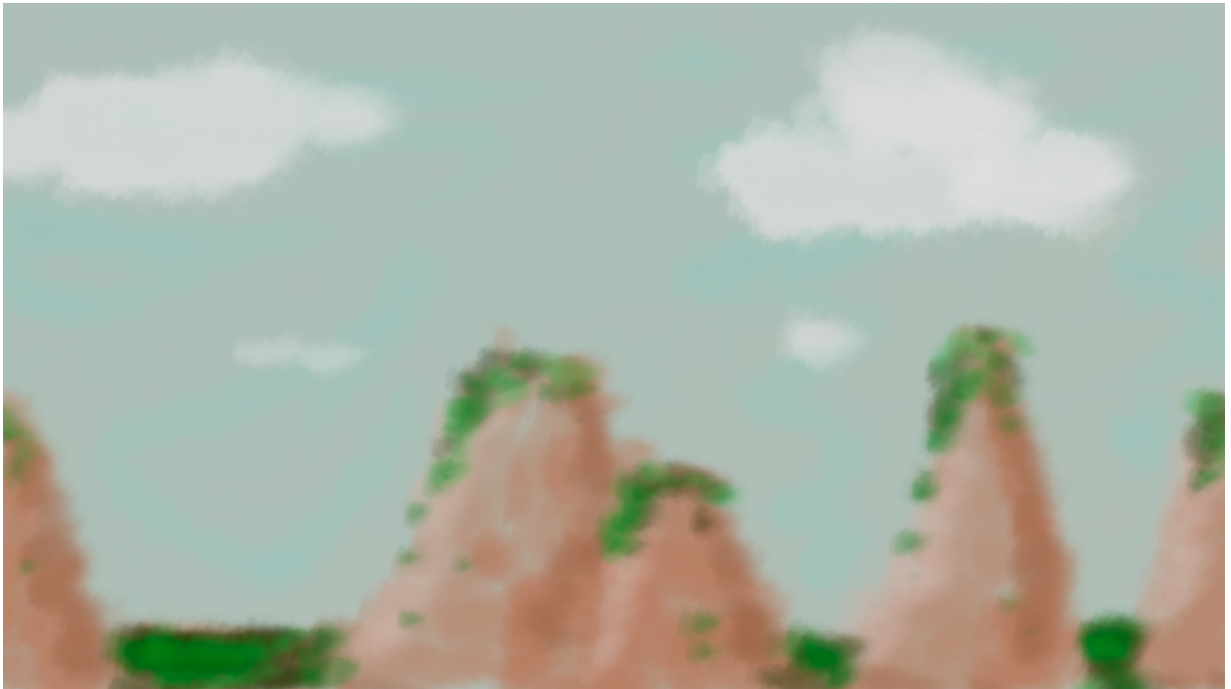


Figure 13: Fondo que aparece de forma aleatoria mientras juegas un minijuego

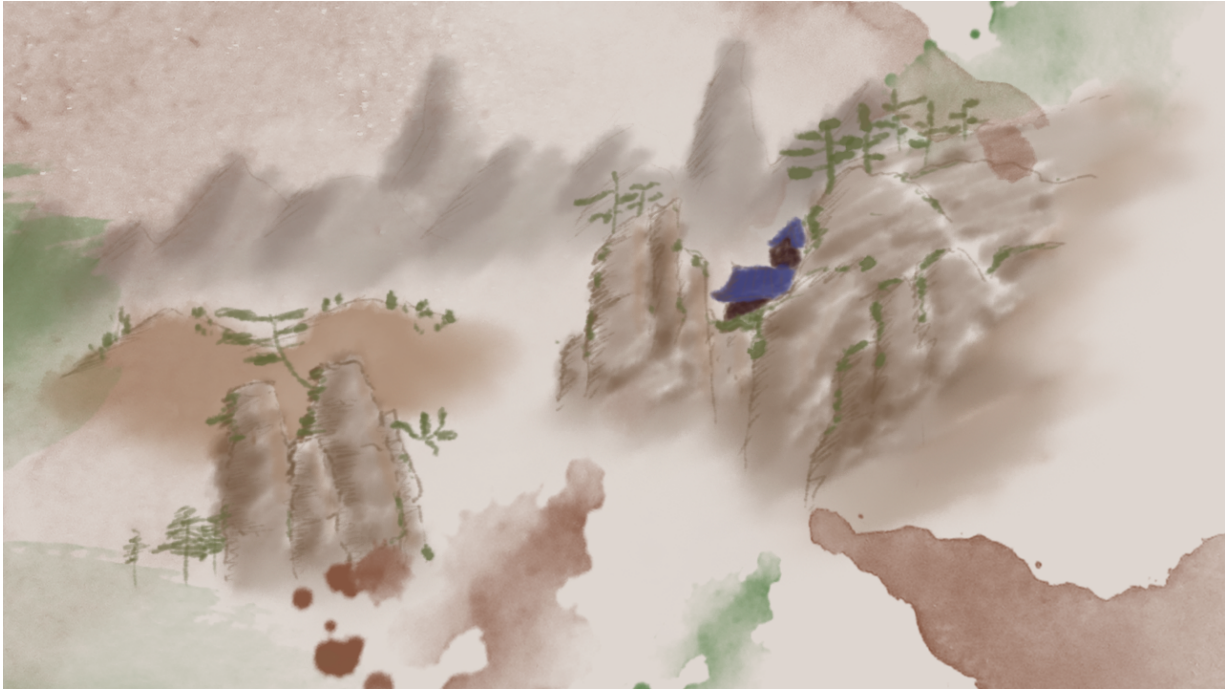


Figure 14: Fondo que aparece de forma aleatoria mientras juegas un minijuego

Por otro lado tenemos la representación de la vida china con las típicas calles tradicionales con muros de piedra y suelos de tierra, además de encontrar diversos puestos de vendedores en las calles.

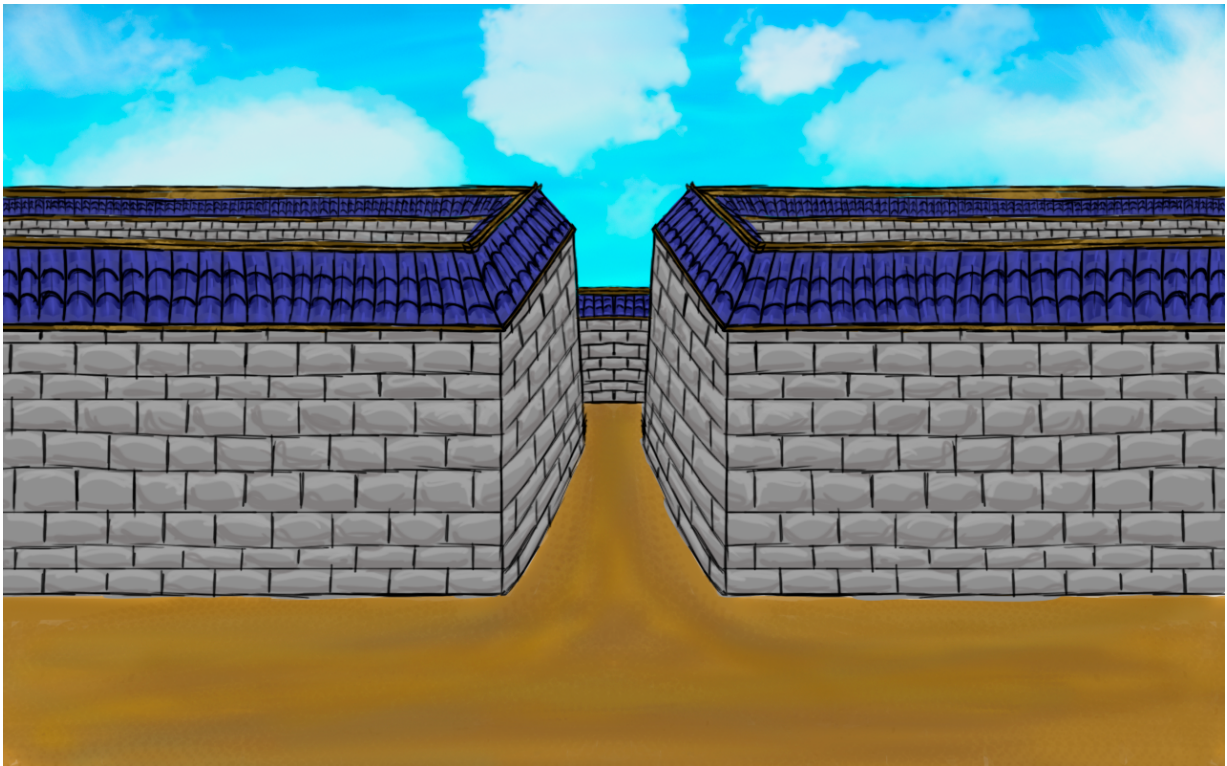


Figure 15: Escenario principal del juego

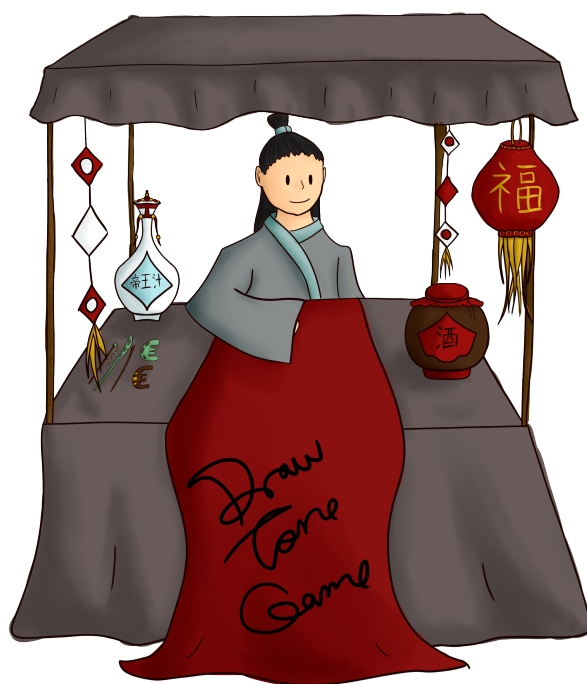


Figure 16: Puesto de vendedor que enlaza con el minijuego de dibujar el tono

5.4 Tecnología

El videojuego está orientado a dispositivos móviles. Una de las tecnologías utilizadas para el desarrollo del juego es un traductor de texto a sonido para obtener el sonido de los distintos hanzi.

Convertor de texto a sonido utilizado es: <https://soundoftext.com>

Una complicación es el desarrollo para teléfonos móviles, ya que he tenido que utilizar el nuevo Input System de Unity del cual no hay tanta información y es más complejo que el anterior.

5.5 Mecánicas

En este apartado explicare las mecánicas principales del juego. Entre ellas el espacio donde sucede el juego, la jerarquía de retos a cumplir por el jugador, los objetos, recursos, atributos y estados que se encuentran en el juego, la relación entre esos objetos y los escenarios; y la economía de la que se nutre el juego.

5.5.1 Espacio de juego

El espacio de nivel de juego a nivel de historia es lineal ya que hay una evolución en las habilidades del jugador, por lo tanto, después de superar el nivel 1, irá al nivel 2 y así sucesivamente. En caso de fallar un nivel, el jugador tiene la posibilidad de repetir ese nivel o volver al anterior para así practicar. Además, el jugador cuenta con el espacio de juego de modo práctica en el que se le permite seleccionar unos parámetros que definen cómo quiere practicar.

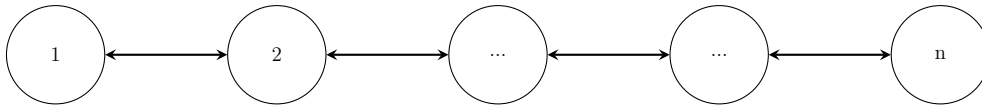


Figure 17: Espacio de juego

Los escenarios son pantallas 2D que su disposición depende enteramente del minijuego o lugar en el que este. Además de un escenario inicial que simula una ciudad China. El jugador puede moverse por 4 escenas principales: la calle principal, la calle trasera (modo práctica), la casa (inventario) y el camino al pueblo (modo historia).

5.5.2 Jerarquía de retos

Para completar el juego, el jugador debe primero conseguir una serie de objetivos más pequeños que se agrupan en objetivos mayores y que al completarlos todos, se completa el juego. Así pues, los objetivos son los siguientes.

- Completar juego
 - Completar tutorial
 - * Completar tutorial de la calle principal
Breve explicación de como funciona la calle principal del juego y cuales son sus características.
 - * Completar tutorial de la calle de práctica
Breve explicación de la calle de práctica del juego y cuales son sus características.
 - * Completar tutorial de la casa
Breve explicación de la casa del juego y cuales son sus características.
 - Completar modo historia
 - * Completar nivel
Para completar un nivel el jugador debe superar el minijuego en cuestión con una cantidad de fallos menos a la permitida (o dentro del tiempo permitido en función del minijuego).
Dada la variedad de minijuegos y que estos se presentan de manera aleatoria, el juego pondrá un tutorial de cada minijuego la primera vez que aparezcan.
Cada nivel primero contiene una explicación del concepto a explicar, después si es necesaria la explicación del minijuego y finalmente el jugador debe superar este minijuego para superar el nivel
 - Completar logros
 - * Superar logro
El juego tiene diversos objetivos secundarios a completar por el jugador, como por ejemplo dominar un número de caracteres, jugar a todos los minijuegos, dominar todos los caracteres...
El jugador debe ir superando estos logros (algunos ocultos y otros no).

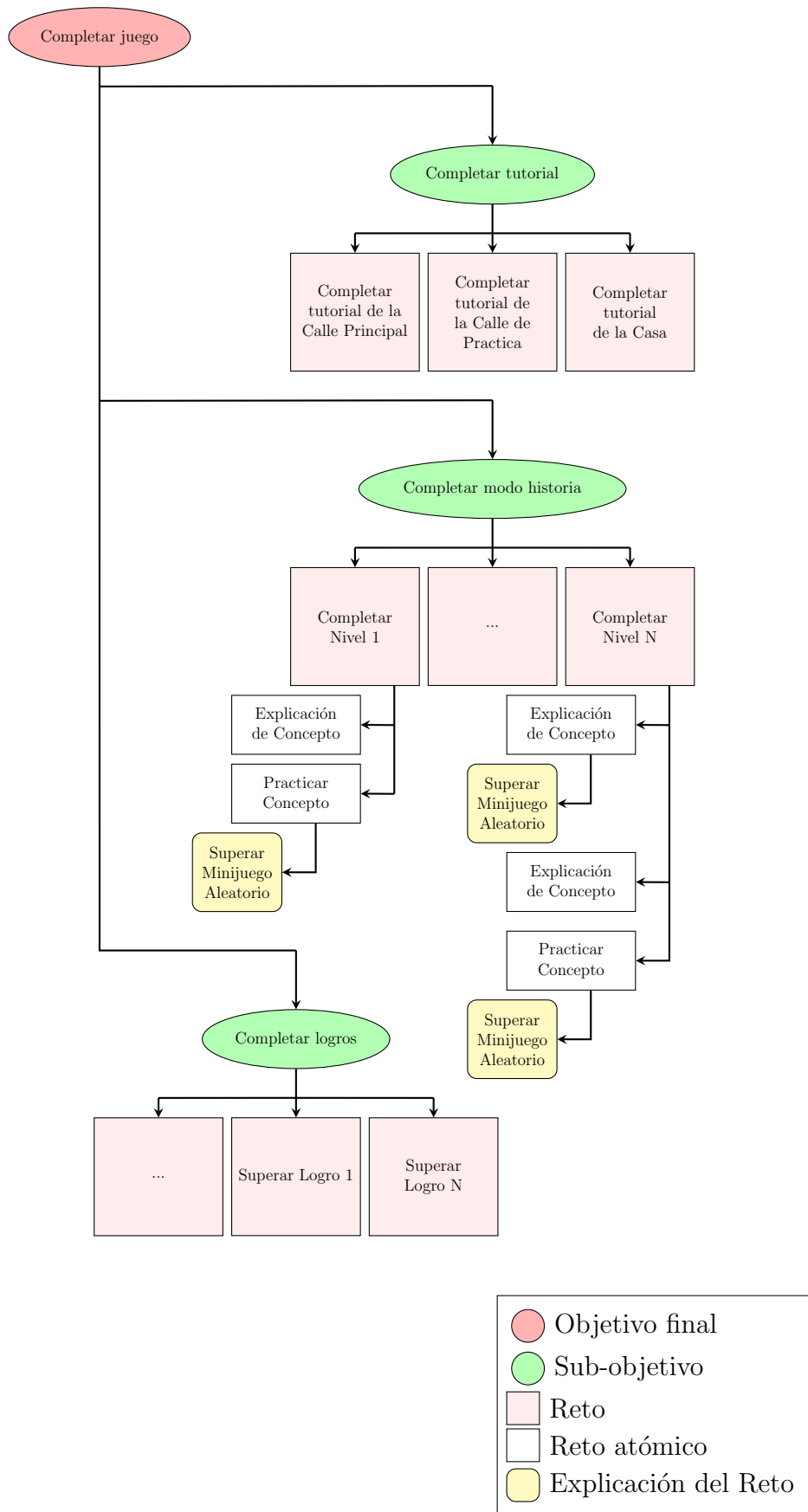


Figure 18: Jerarquía de retos global. Visión de los objetivos principales del juego.

5.5.3 Objetos, recursos, atributos y estados

En esta sección veremos los objetos, recursos, atributos, estados y la relación entre estos además de sus funciones asociadas. Así pues, las tablas muestran los objetos con sus correspondientes atributos y bajo estas se encuentran las funciones que aplican.

1. Recursos y funciones asociadas

| Caracter | |
|---------------------------|-------------------------|
| Caracter | <i>(Carácter chino)</i> |
| Sonido | <i>(Sonido)</i> |
| Pinyin | <i>(Pinyin)</i> |
| Número de trazos | <i>1-n</i> |
| Traducción | <i>(Traducción)</i> |
| Porcentaje de aprendizaje | <i>0-100%</i> |

- `AumentarPorcentaje()`
Aumenta el porcentaje de completado del .
- `DisminuirPorcentaje()`
Disminuye el porcentaje de completado del caracter.

| Tarjeta de Memoria | |
|--------------------|---|
| Tipo | <i>Pinyin, Traducción, Sonido, Carácter</i> |
| Caracter | <i>(Carácter chino)</i> |

- `voltear()`
Al hacer click en una tarjeta de memoria, la voltea mostrando su otra cara: pinyin, traducción, sonido o hanzi.
- `mostar()`
Genera la carta boca a bajo, es decir sin mostrar su pinyin, traducción sonido o hanzi.

| Tarjeta Pinyin + Sonido | |
|-------------------------|-----------------|
| Pinyin | <i>(Pinyin)</i> |
| Sonido | <i>(Sonido)</i> |

- `mostrar()`
Genera la tarjeta que muestra reproduce el sonido del pinyin, y el pinyin de esta.

| Dibujo Carácter | |
|------------------|-------|
| Número de trazos | $1-n$ |
| Errores | $0-n$ |

- `corregirTrazos()`

Muestra los trazos a dibujar en el orden correcto ya que el jugador se ha equivocado.

| Dibujo Tono | |
|-------------|--------------|
| Correcto | <i>Si-No</i> |

- `correcto()`

Muestra si el tono dibujado por el jugador es correcto o no, en caso de no serlo muestra el correcto.

| Tarjeta Carácter + Pinyin + Sonido | |
|------------------------------------|-------------------|
| Carácter | <i>(Carácter)</i> |
| Pinyin | <i>(Pinyin)</i> |
| Sonido | <i>(Sonido)</i> |

- `mostrar()`

Muestra la tarjeta con el carácter y el pinyin y reproduce su sonido.

| Tarjeta nivel | |
|---------------|--------------------|
| Nivel | <i>(Aleatorio)</i> |

- `generar()`

Genera automáticamente una tarjeta de nivel teniendo en cuenta que el número del nivel será igual a el número de caracteres desbloqueados más uno. Además, esta tarjeta enlaza automáticamente a un minijuego aleatorio que se decide cada vez que se selecciona.

- `mostrar()`

Muestra la tarjeta de nivel generada.

| Tarjeta de Sonido | |
|-------------------|-----------------|
| Sonido | <i>(Sonido)</i> |
| Número de trazos | <i>1-n</i> |

- mostrar()
Reproduce el sonido del caracter.

| Tarjeta de Orden | |
|------------------|--------------------|
| Carácter | <i>(Cáacter)</i> |
| Letras | <i>(Aleatoria)</i> |

- generar()
Genera elementos seleccionables de forma aleatoria que contienen letras (entre ellas siempre se encontraran las letras que compoenn el pinyin del caracter).
- mostrar()
Muestra el resultado correcto de la tarjeta.

2. Objetos

Estos son algunos ejemplos de objetos. No incluyo todos dado que son demasiados.

| Caracter | |
|---------------------------|-----------------|
| Caracter | 一 |
| Sonido | <i>(sonido)</i> |
| Pinyin | <i>(yī)</i> |
| Número de trazos | 1 |
| Traducción | <i>Uno</i> |
| Porcentaje de aprendizaje | 0% |

| Caracter | |
|---------------------------|-----------------|
| Caracter | 儿 |
| Sonido | <i>(sonido)</i> |
| Pinyin | <i>(ér)</i> |
| Número de trazos | 2 |
| Traducción | <i>Piernas</i> |
| Porcentaje de aprendizaje | 0% |

| Caracter | |
|---------------------------|-----------------|
| Caracter | 心 |
| Sonido | <i>(sonido)</i> |
| Pinyin | <i>(xīn)</i> |
| Número de trazos | 4 |
| Traducción | <i>Corazón</i> |
| Porcentaje de aprendizaje | 0% |

| Caracter | |
|---------------------------|-----------------|
| Caracter | 龠 |
| Sonido | <i>(sonido)</i> |
| Pinyin | <i>(yuè)</i> |
| Número de trazos | 17 |
| Traducción | <i>Flauta</i> |
| Porcentaje de aprendizaje | 0% |

| Tarjeta de Memoria | |
|--------------------|---------------|
| Tipo | <i>Pinyin</i> |
| Caracter | 心 |

| Tarjeta de Memoria | |
|--------------------|-------------------|
| Tipo | <i>Traducción</i> |
| Caracter | 心 |

| Tarjeta de Memoria | |
|--------------------|---------------|
| Tipo | <i>Sonido</i> |
| Caracter | 心 |

| Tarjeta de Memoria | |
|--------------------|-----------------|
| Tipo | <i>Carácter</i> |
| Caracter | 心 |

3. Relación Escenarios-Objetos

La relación escenarios-objetos depende íntegramente del minijuego por lo que hablaré de relación minijuego-objetos. Partiendo de la base de que el jugador posee un "inventario" con todos los caracteres.

- Minijuego Memoria
Durante este minijuego habrán diversas tarjetas de memoria de diversos tipos que incluirán los caracteres ya aprendidos anteriormente.
- Minijuego Detector de Errores
Durante este minijuego habrá una tarjeta Pinyin + Sonido de el carácter que este aprendiendo o repasando
Además durante este minijuego el objeto "Dibujo Carácter" irá actualizando el número de errores detectados.
- Minijuego Dibujar el Tono
Durante este minijuego habrá una tarjeta Carácter + Pinyin + Sonido de un único Carácter cada vez (hasta acabar el juego se generarán más tarjetas).
Existe además el objeto Dibujo Tono que indica si el tono se ha dibujado correctamente o no.
- Minijuego Unir
Se muestra una única Tarjeta de nivel en la que se generan aleatoriamente caminos hacia los radicales para que el jugador los coloque.
- Minijuego Gacha
Se muestra una única Tarjeta de nivel en la que se genera aleatoriamente Tarjetas de Caracteres que el jugador debe mover al tono correcto.
- Minijuego Escucha
Se muestra una única Tarjeta de nivel (se generan más hasta acabar el minijuego). El jugador debe seleccionar la Tarjeta de radical correcta mostrada que se corresponda con el sonido escuchado.
- Minijuego Dibuja Escucha
Se muestra la Tarjeta de Sonido única por minijuego y el jugador Dibuja el Carácter.
- Minijuego Ordenar Pinyin
Se muestra la Tarjeta de Orden del minijuego y el jugador coloca en orden el pinyin.

4. Acciones

A continuación se presentan las diferentes acciones que se pueden realizar y una pequeña explicación de estas.

AumentarPorcentaje(x) { porcentaje += x; Aumenta el porcentaje de un elemento. }

DisminuirPorcentaje(x) { porcentaje -= x; Disminuye el porcentaje de un elemento. }

voltear() { voltea la Tarjeta y si es igual a la otra tarjeta volteada lo indica. }

mostrar() { muestra la tarjeta (o nivel) y todo su contenido }

corregirTrazos(fallos) { Errores += fallos muestra Trazos correctos }

generar() { genera un nivel aleatorio basado en unos parámetros base }

5.5.4 Economía del juego

Los pilares de la economía de todo juego son los sources, drains, converters y traders. Los sources son las principales fuentes que generar un recurso. Los drains son los que disminuyen un recurso. Los converters convierten un recurso en otro permanentemente. Los traders intercambian un recurso por otro.

En el caso de este videojuego nos encontramos con dos recursos principales: el porcentaje de progreso de un carácter y el porcentaje de progreso de un token.

En el caso de los caracteres son recursos los cuales aumentas y disminuyes jugando a los minijuegos, es decir que los sources y drains estan implícitos en los minijuegos en función de los errores y aciertos del jugador.

En el caso de los tokens son recursos que aumentan y disminuyen a la par que los caracteres (pero más lentamente) mediante los sources y drains que son los aciertos y fallos de los jugadores en los minijuegos. Además de poseer la especial característica de poder convertir porcentaje de progreso de un caracter en porcentaje de progreso de un token en una proporción 10:1. También son traders ya que puedes intercambiar el porcentaje de un token a otro proporción 1:1.

- Funciones de la economía del juego

- Carácter

- * AumentarPorcentaje(x);

- * DisminuirPorcentaje(x);

- Token

- * Recompensa();

- * AumentarGrupo(x,grupo);

- * DisminuirGrupo(x,grupo);

- * ConvertirPorcentajeDelCaracterEnGrupo(caracter,grupo);

- * ConveterPorcentajeDeUnGrupoEnOtroGrupo(grupo1,grupo2);

Los Sources que encontramos en este videojuego serían AumentarPorcentaje(x); y AumentarGrupo(x,grupo); Estos Sources convierten los aciertos de un jugador durante un minijuego en porcentaje de progreso del caracter en cuestión.

Los Drains serían DisminuirPorcentaje(x); y DisminuirGrupo(x); Estos Drains convierten los fallos de un jugador durante un minijuego en una disminución de porcentaje de progreso del caracter en cuestión.

El Converter que encontramos sería ConvertirPorcentajeDelCaracterEnGupo(caracter,grupo); Este Converter convierte un 10% del progreso de un carácter en un 1% de progreso de un Token de manera permanente sin la posibilidad de recuperarlo.

El Trader sería ConveterPorcentajeDeUnGrupoEnOtroGrupo(grupo1,grupo2); Este Trader convierte un 10% de porcentaje de un Token en otro Token. En esta conversión, el recurso no se pierde ya que siempre se puede convertir de vuelta.

El juego brindará al jugador de otra manera de obtener estos tokens de forma aleatoria. Al acabar un minijuego de forma perfecta (sin fallos), se le dará al jugador una bolsa aleatoria de tokens y cada token de esta bolsa puede ser de cualquier grupo de tokens. Siendo esto un Source de Tokens asociado a la función Recompensa();

5.6 Interfaces

- Pantalla de Inicio

Es la pantalla inicial del videojuego que aparece nada más iniciar el videojuego.

1. Opciones

Permite al jugador configurar las diferentes opciones del dispositivo.

2. Continuar Partida

Permite al jugador continuar con una partida ya empezada.

3. Nueva Partida

Permite al jugador empezar una nueva partida.

4. Salir

Permite al jugador salir del juego.

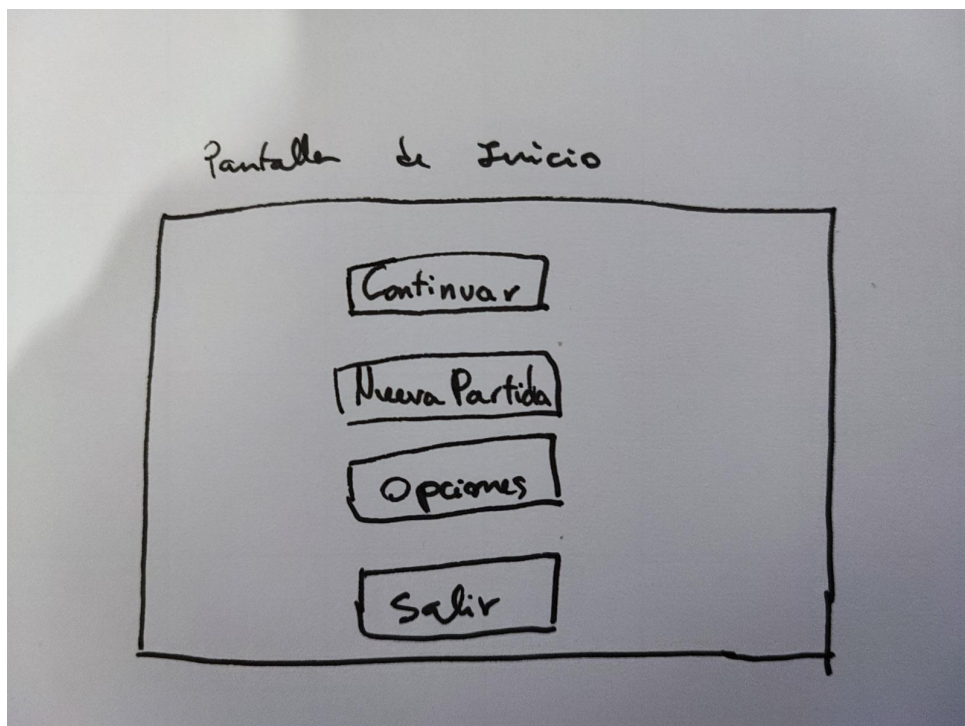


Figure 19: Esquema del menú al iniciar el juego

- Opciones

Es la pantalla de opciones que permite realizar diversas configuraciones del videojuego.

1. Video

Opciones de configuración de video.

2. Audio

Opciones de configuración de audio.

3. Volver

Vuelve a la pantalla de inicio.

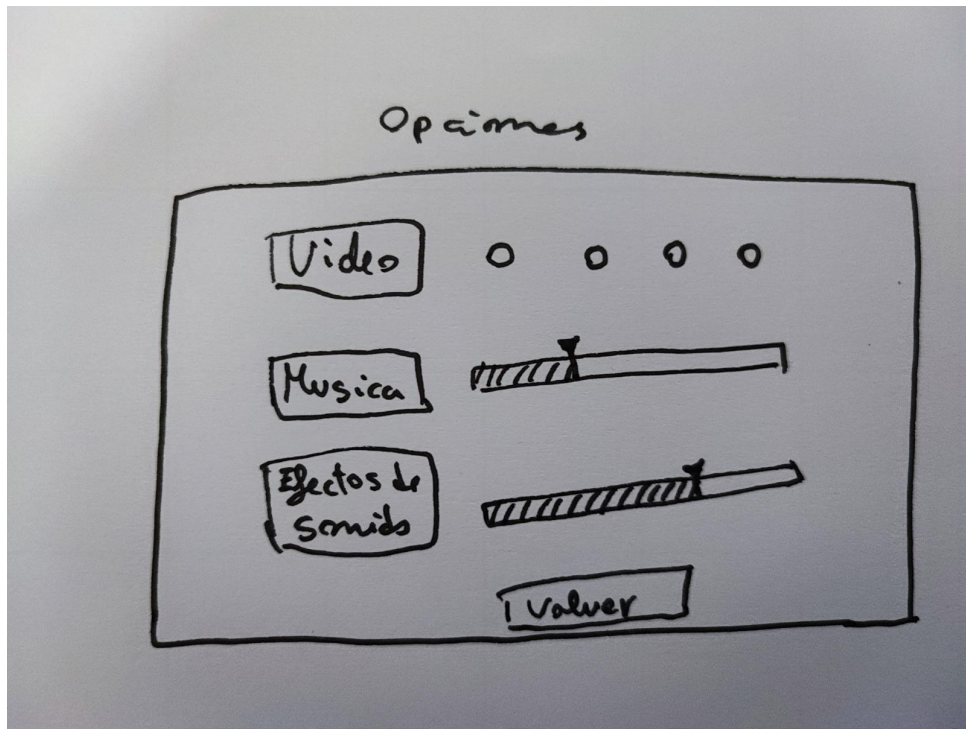


Figure 20: Esquema del menú al iniciar el juego

- Mundo de Juego

El mundo de juego es la zona donde el jugador puede seleccionar los distintos modos de juego o ir a su casa.

1. Modo Práctica

Lleva al jugador a la pantalla de modo práctica.

2. Casa

Lleva al jugador a la pantalla de casa.

3. Modo Historia

Lleva al jugador a la pantalla de modo historia.

4. Volver al Inicio

Vuelve a la pantalla de inicio.

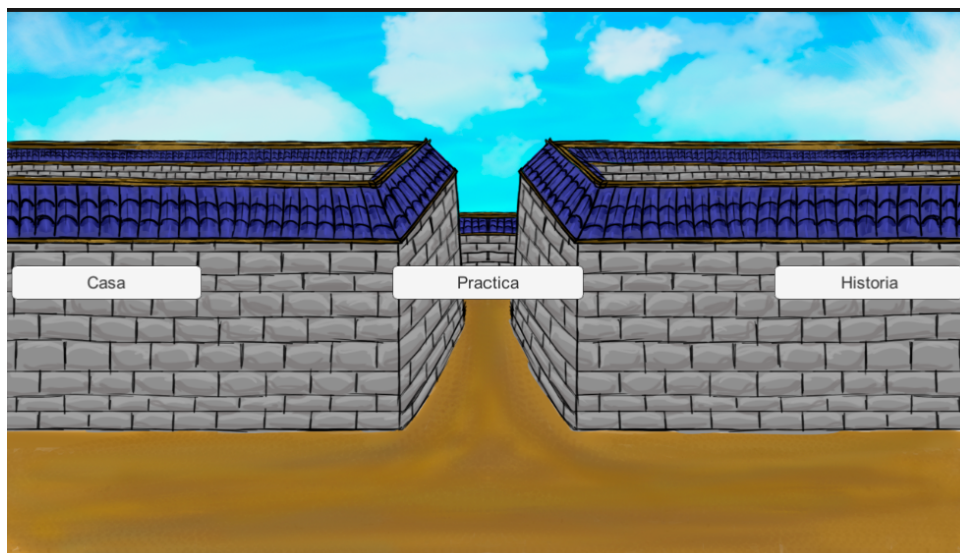


Figure 21: Esquema del menú al iniciar el juego

- Pantalla de Feedback de los Minijuegos

Al acabar los minijuegos aparecen las patallas de feedback en las que se le muestra al jugador su rendimiento además de diversas opciones.

1. Volver

Lleva al jugador a la pantalla de modo práctica si se accedió al minijuego a través del modo práctica o al modo historia si se accedió al minijuego a través del modo historia.

2. Recompensa

Aparece la recompensa si es el momento como un elemento interctuable.



Figure 22: Esquema del menú al iniciar el juego

- Casa

El jugador tiene la opción de acceder a su casa desde el mundo principal de juego para visualizar su progreso en los caracteres y acceder a los Tokens.

1. Volver

Lleva al jugador a la pantalla del mundo de juego.

2. Sacos

Lleva al jugador a la pantalla de tokens.

3. Carácteres

Un menú desplazable hacia la derecha y la izquierda en el cual se encuentran los caracteres que al hacerles click se escucha su sonido y si se ve su porcentaje de completado.



Figure 23: Esquema del menú al iniciar el juego

- Pantalla de Tokens

El jugador puede ver sus tokens e interactuar con ellos además de acceder al examen final.

1. Examen

Lleva al jugador a la pantalla de examen.

2. Casa

Lleva al jugador a la pantalla de casa.

3. Tokens

Un menú desplazable hacia la derecha y la izquierda en el cual se encuentran los tokens y se muestra su porcentaje de completado. Al hacerles click aparece la pantalla que permite al jugador convertir porcentaje de caracteres o de otros tokens en porcentaje del seleccionado.

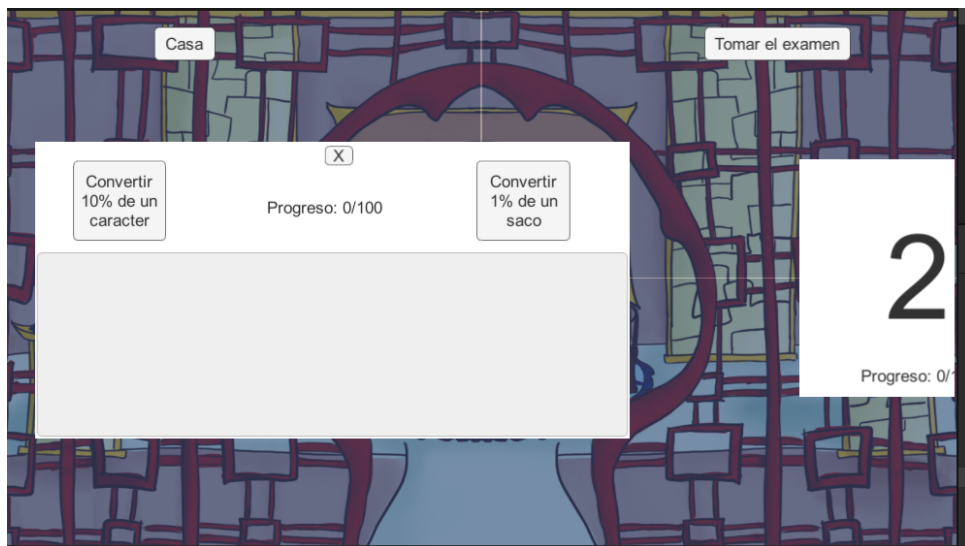


Figure 24: Esquema del menú al iniciar el juego

- Pantalla de Examen

En cualquier momento, desde la pantalla de tokens, el jugador puede acceder a la pantalla de examen para poner a prueba sus conocimientos.

1. Casa

Lleva al jugador a la pantalla de casa.

2. Comprobar

Comprueba si la frase escrita por el jugador es la solicitada.

3. Tokens

Un menú desplazable hacia la derecha y la izquierda en el cual se encuentran los tokens. Al hacerles click aparece otro menú desplazable que permite al jugador seleccionar el caracter a escribir.



Figure 25: Esquema del menú al iniciar el juego

- Pantalla de Historia

A través del mundo de juego se puede acceder a esta pantalla la cual te permite jugar a los diversos minijuegos y desbloquear nuevos caracteres.

1. Volver

Lleva al jugador a la pantalla de mundo de juego.

2. Niveles

Un menú desplazable hacia la derecha y la izquierda en el cual se encuentran los diversos niveles, tanto los desbloqueados como los que no. Al hacer click en uno de estos te llevará a un minijuego aleatorio.

Continúa al siguiente nivel.



Figure 26: Esquema del menú al iniciar el juego

- Pantalla de Práctica

A través del mundo de juego se puede acceder a esta pantalla la cual te permite jugar a los diversos minijuegos.

1. Volver

Lleva al jugador a la pantalla de mundo de juego.

2. Minijuegos

Un menú desplazable hacia la derecha y la izquierda en el cual se encuentran los diversos minijuegos. Al hacer click en uno de estos te llevará al minijuego seleccionado.

- Pantalla de Selector de dificultad

A través del modo practica se puede acceder a esta pantalla la cual te permite seleccionar la dificultad y el conjunto de caracteres con el que jugar al minijuego seleccionado.



Figure 27: Esquema del menú al iniciar el juego

1. Dificultad

Con opciones

- (a) Fácil
- (b) Media
- (c) Difícil

2. Conjunto de caracteres

(a) Peores

Los peores caracteres que lleva el jugador, es decir con porcentaje por debajo del 50%.

(b) Seleccionar

Un menú desplazable hacia la derecha y la izquierda en el cual se encuentran los diversos caracteres y el jugador los puede añadir a una lista seleccionandolos.

(c) Aleatorio

Un conjunto aleatorio de caracteres. (Esta opción es la por defecto en el modo historia).



Figure 28: Esquema del menú al iniciar el juego

5.7 Game layout charts

En esta sección se mostrará la estrategia de diseño a seguir y los diagramas de módulos

5.7.1 Estrategia de diseño a seguir

La estrategia de diseño a seguir es el estilo Bottom-up: partimos de que el jugador no tiene una base en el tema en cuestión y le enseñamos como desarrollar sus habilidades a través de los diversos minijuegos y la repetición de estos.

5.7.2 Diagramas de módulos

A continuación se mostrarán los diagramas de módulo que sigue el juego, partiendo de una visión global del juego a una más detallada de cada sección y cada minijuego.

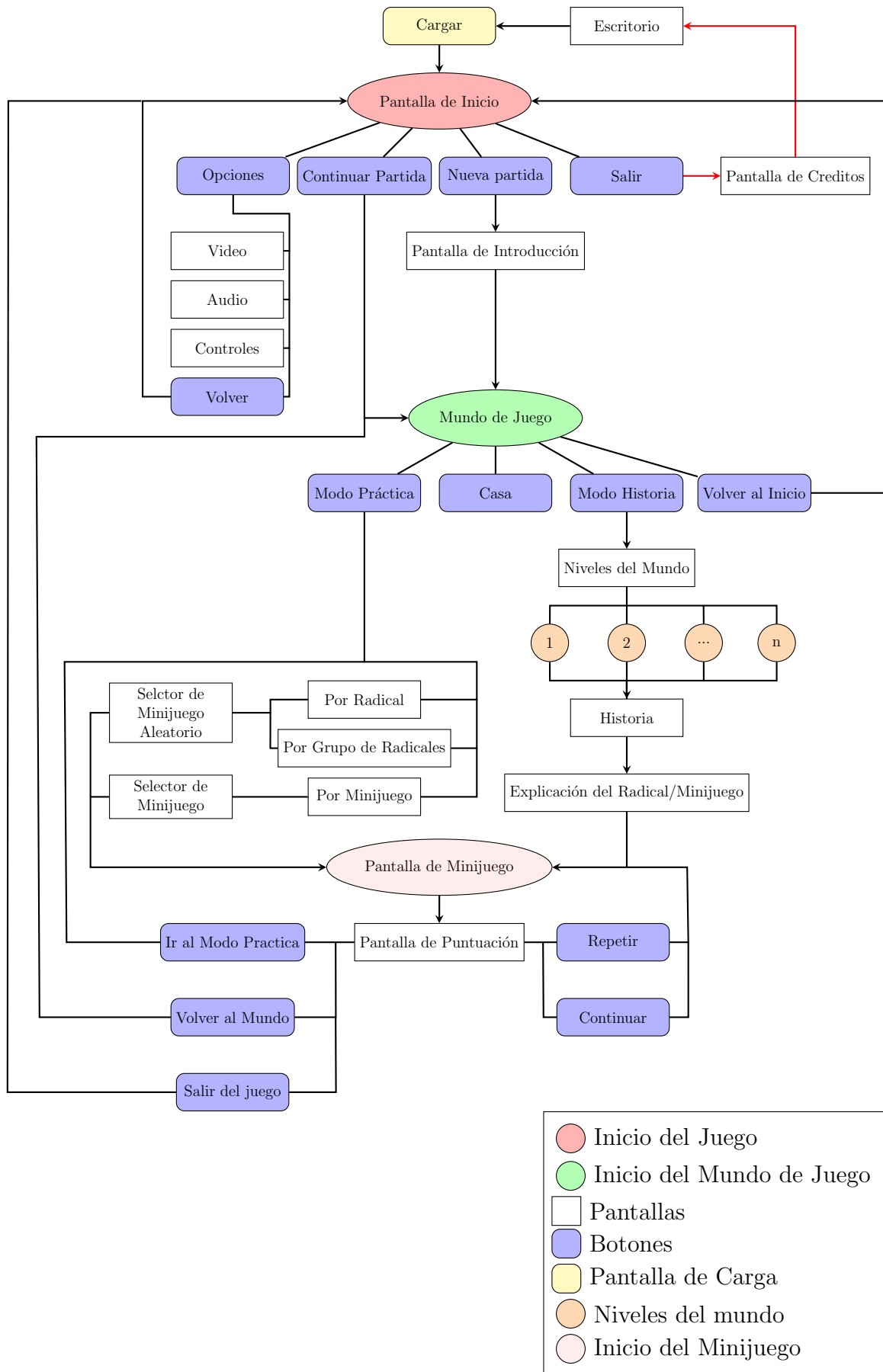


Figure 29: Visión global del juego. Explicación del funcionamiento de la aplicación desde que se abre hasta que se cierra la aplicación.

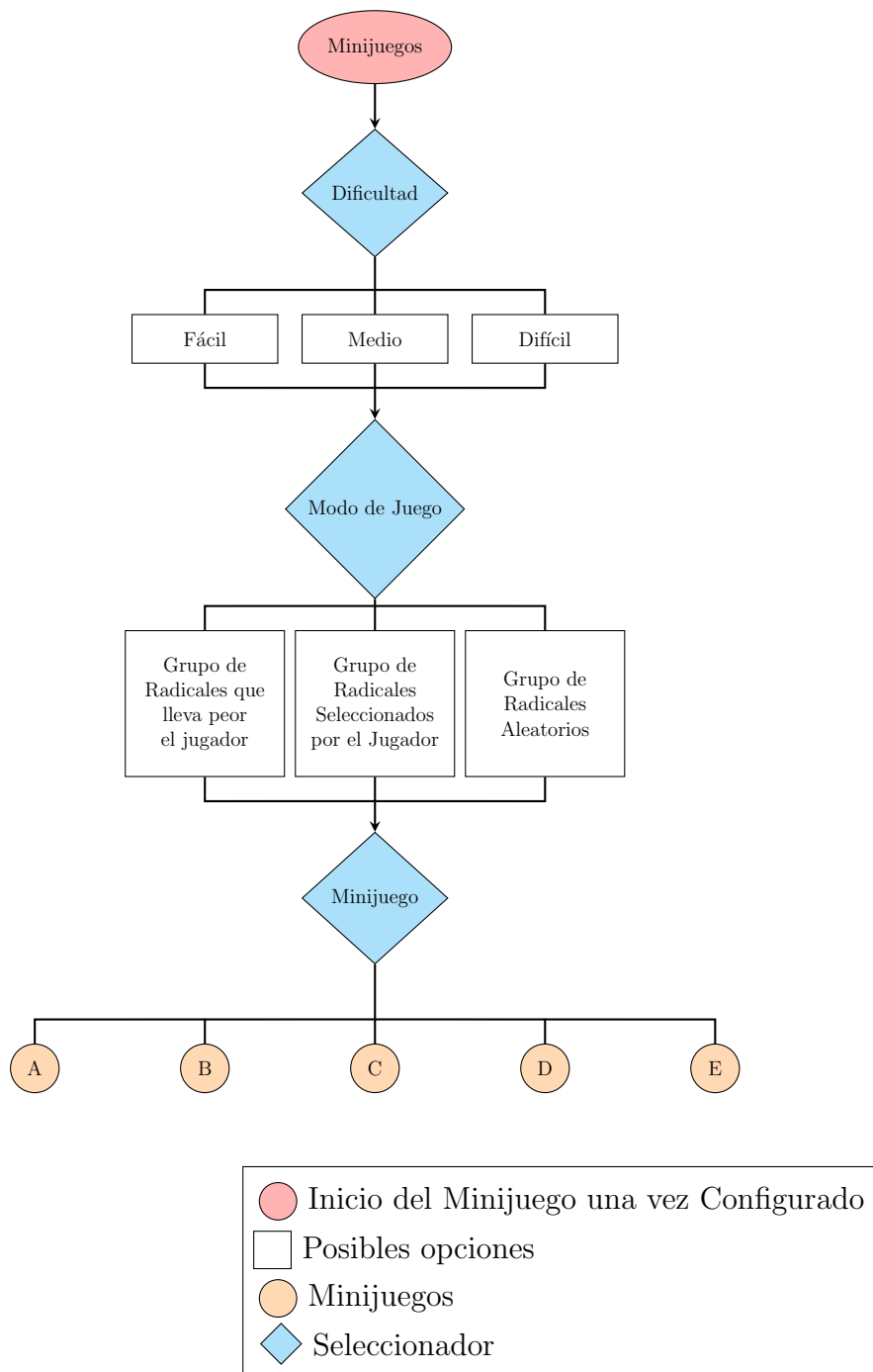


Figure 30: Visión global del funcionamiento de los minijuegos. Explicación del funcionamiento de la selección de dificultad y el grupo de Radicales de los minijuegos.

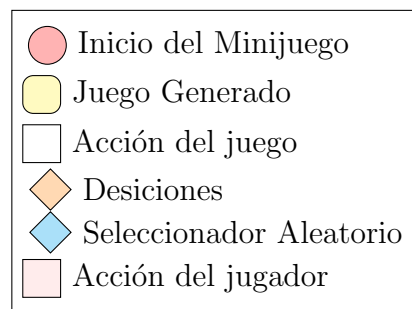
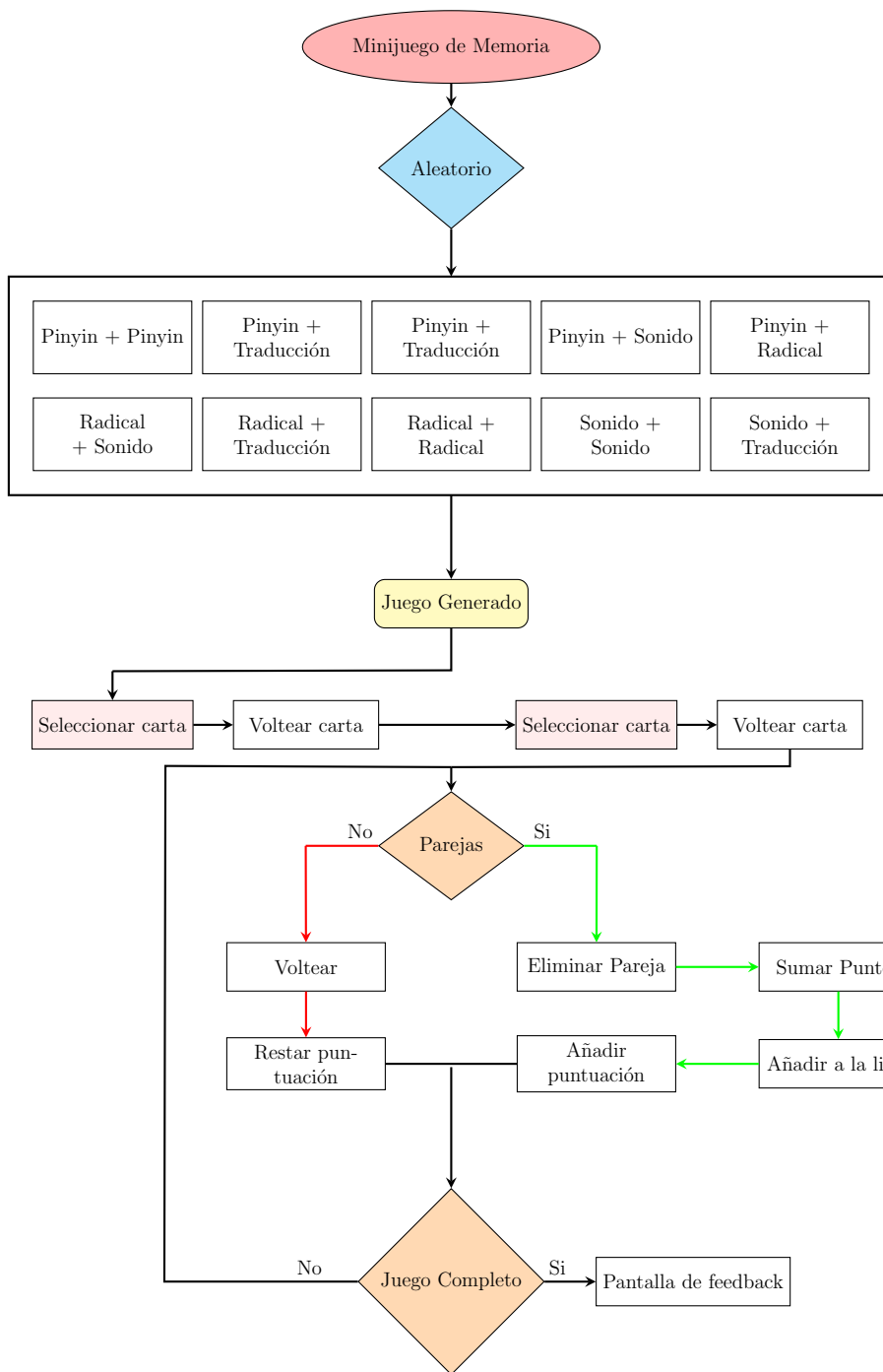


Figure 31: Visión minijuego de Memoria. Explicación del minijuego de memoria desde que se inicia hasta que se completa.

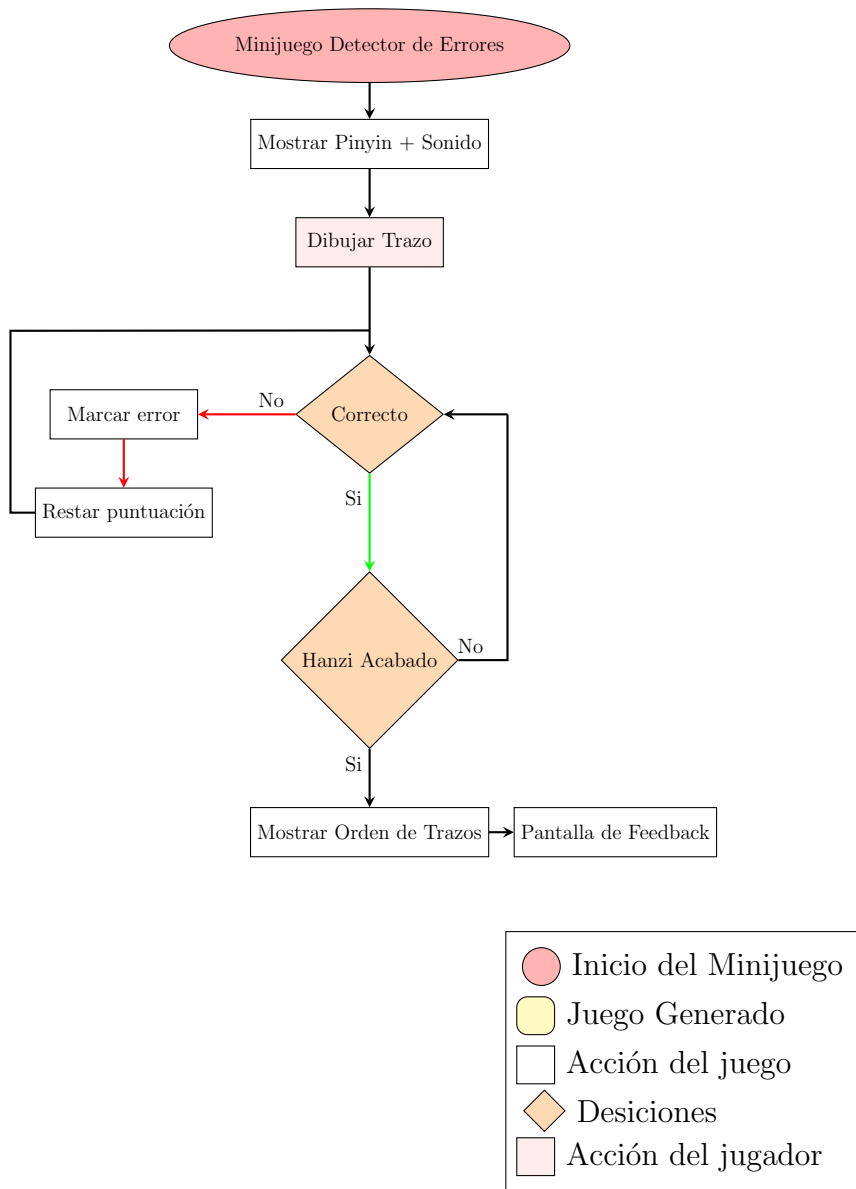


Figure 32: Visión minijuego de Detección de Errores. Explicación del minijuego de detección de errores desde que se inicia hasta que se completa

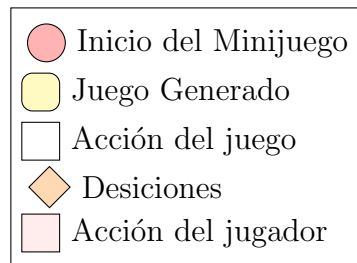
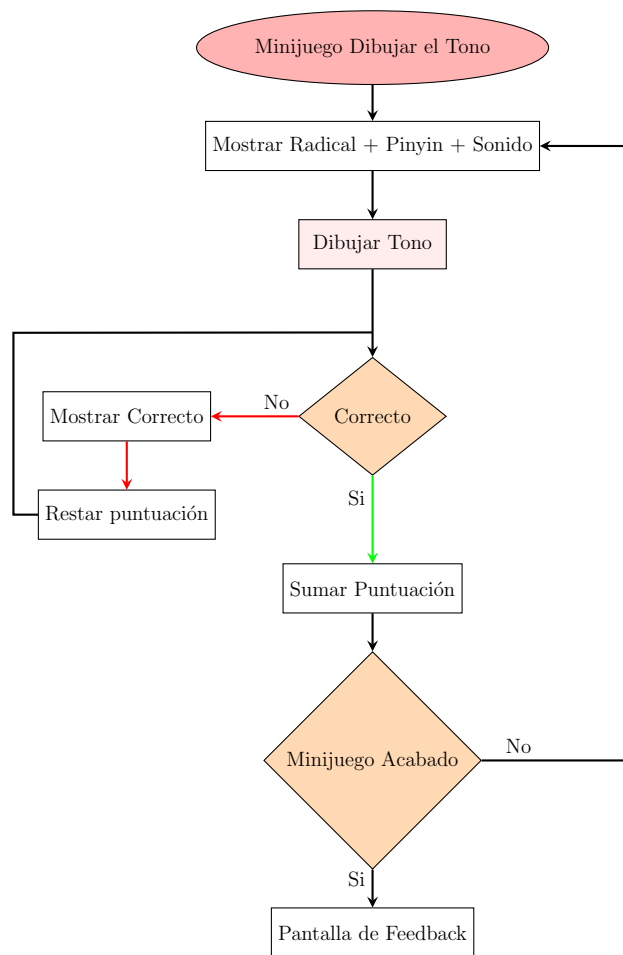


Figure 33: Visión minijuego de Dibujar el Tono. Explicación del minijuego de dibujar el tono desde que se inicia hasta que se completa

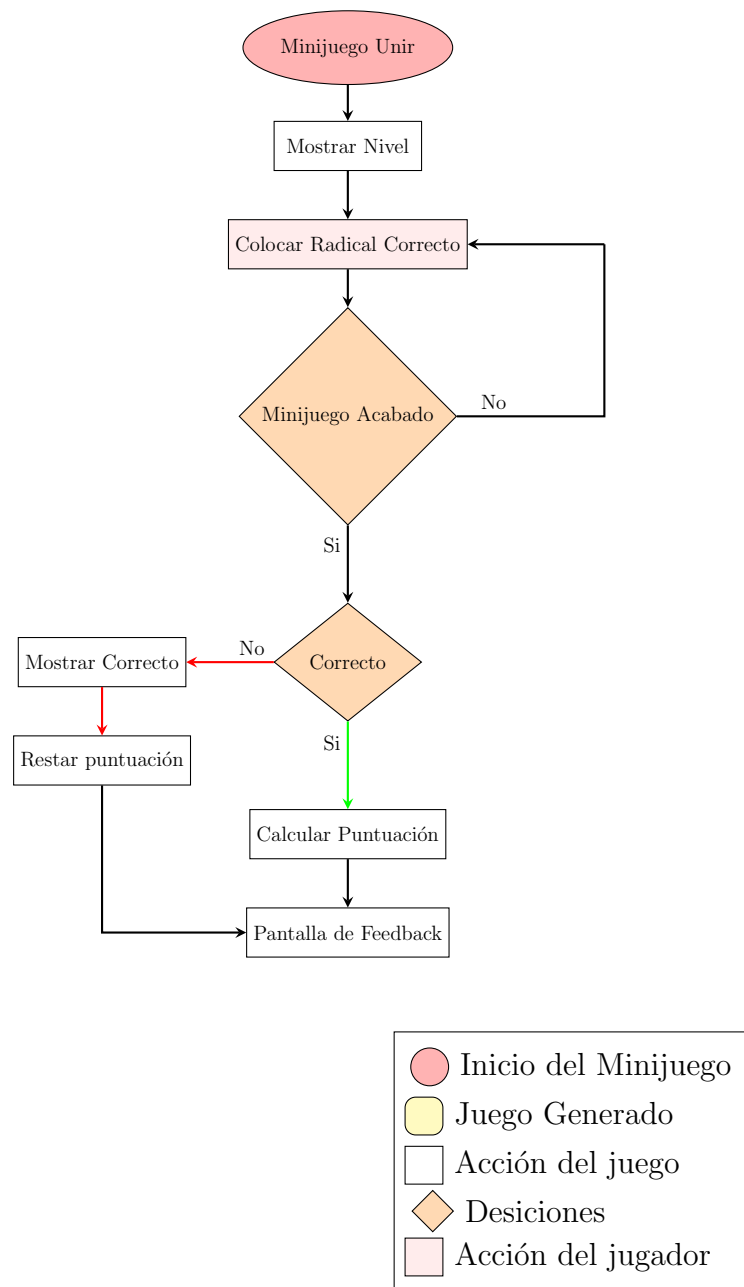


Figure 34: Visión Minijuego de Unir. Explicación del minijuego de unir radical con su pinyin desde que se inicia hasta que se completa

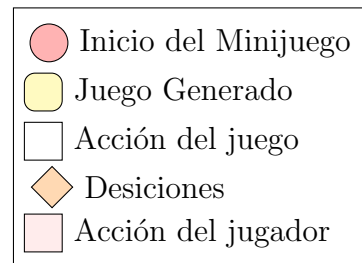
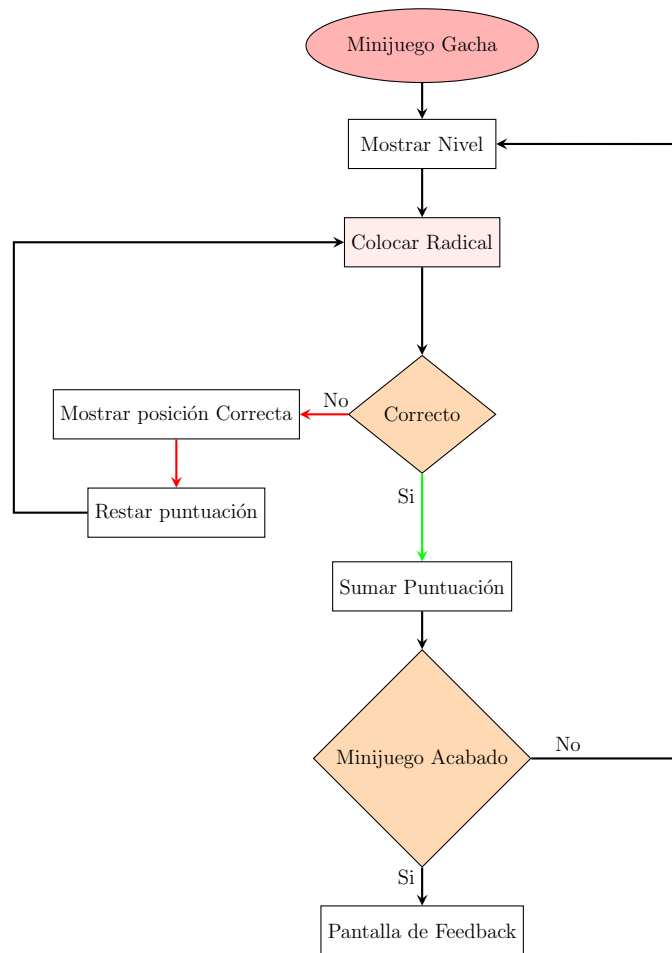


Figure 35: Visión minijuego de Gacha. Explicación del minijuego de gacha desde que se inicia hasta que se completa

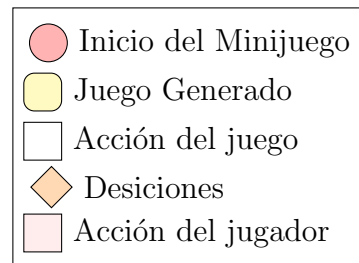
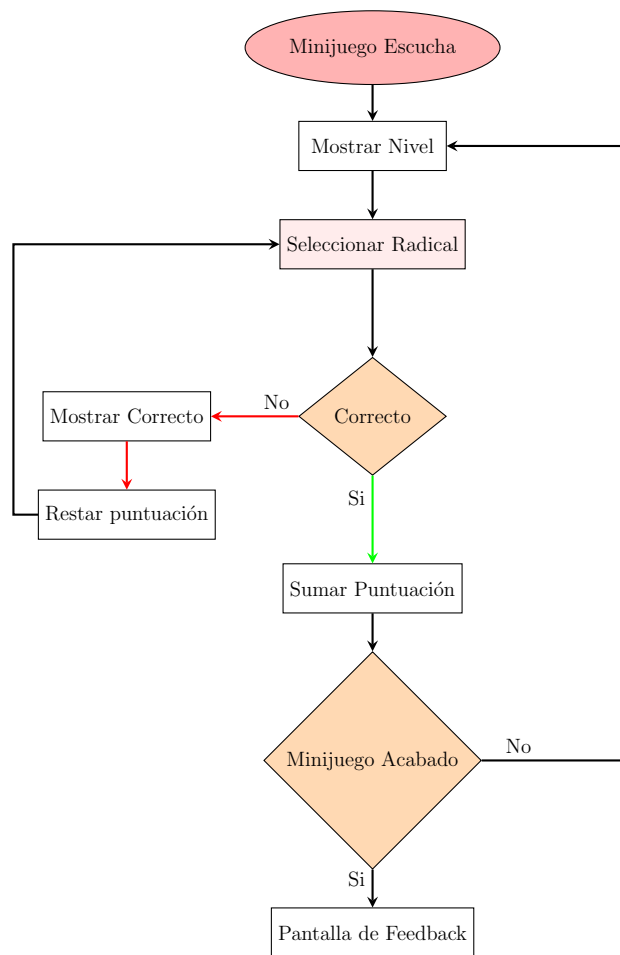


Figure 36: Visión minijuego de Escucha. Explicación del minijuego de escucha desde que se inicia hasta que se completa

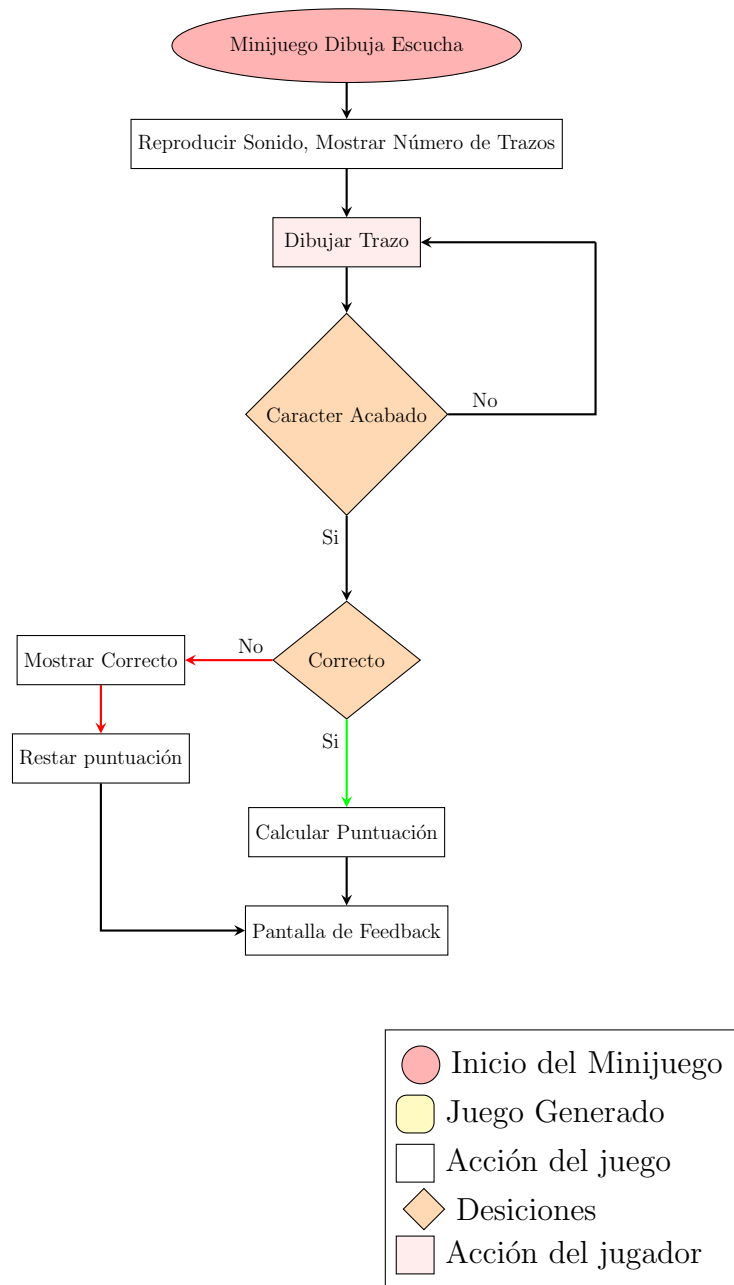


Figure 37: Visión minijuego de Dibuja Escucha. Explicación del minijuego de dibujar lo escuchado desde que se inicia hasta que se completa

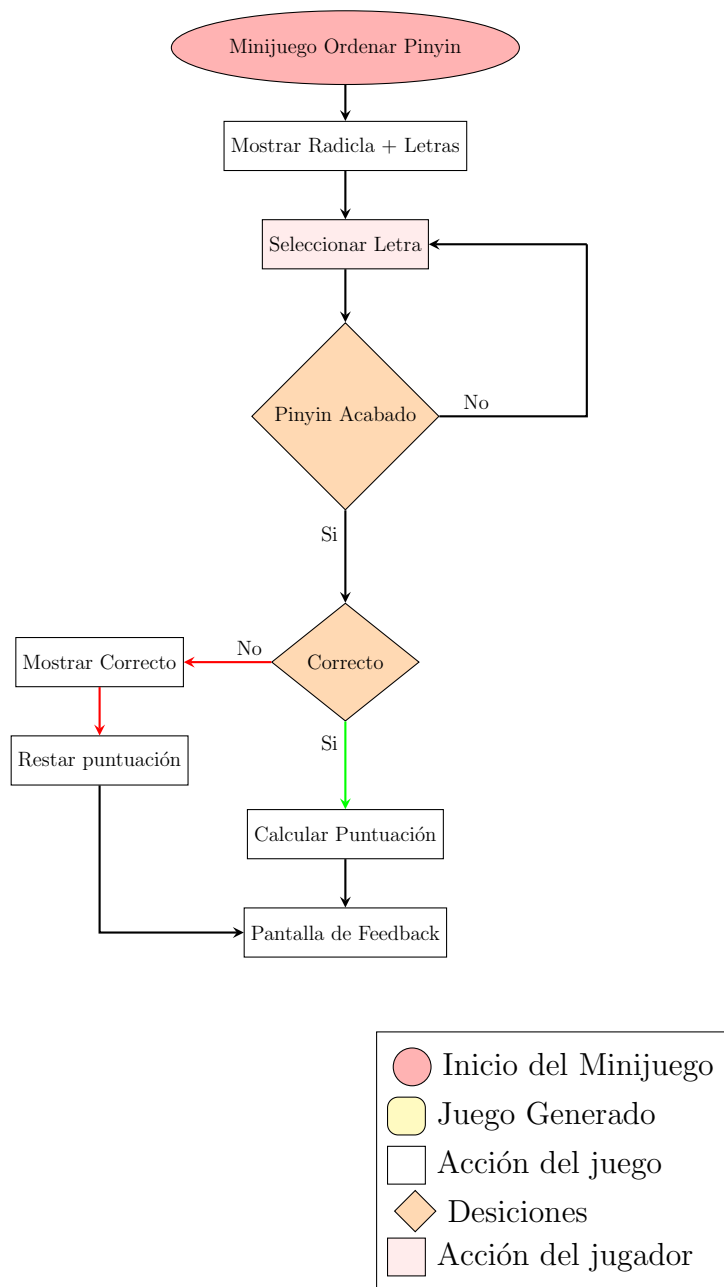


Figure 38: Visión minijuego de Ordenar Pinyin. Explicación del minijuego de ordenar el pinyin del radical desde que se inicia hasta que se completa

6 Implementación

La implementación del juego podría dividirse en cuatro grandes secciones: la base de datos, los menús, los inputs y los minijuegos.

6.1 Base de datos

A continuación muestro un esquema para ver las relaciones que existen en esta simple base de datos. Dentro del apartado de base de datos encontramos dos elementos principales los cuales

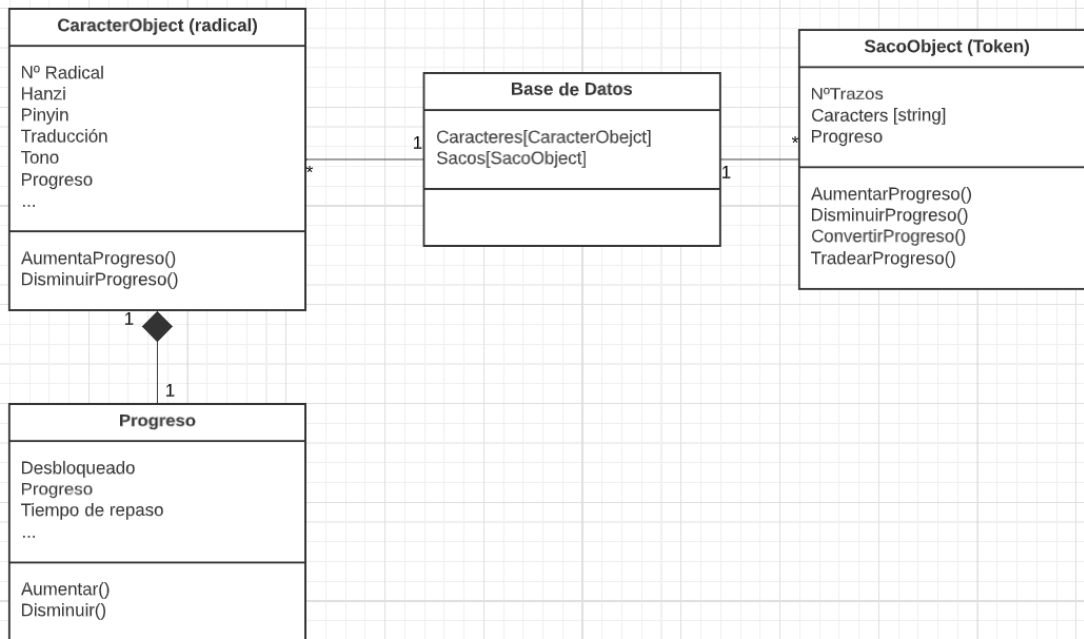


Figure 39: Esquema general de la base de datos

ambos son scriptables object (es decir, una clase que permite almacenar grandes cantidades de datos compartidos independientes de las instancias de los scripts) y se almacenan en listas que son comunicadas atavés de los objetos y escenas.

Por un lado tenemos el objeto `Caracter`, que es el objeto más complejo ya que esta lleno de datos importantes como pueden ser el hanzi del caracter, su pinyin, sus correspondientes imágenes...

```
[CreateAssetMenu(fileName = "New Character Object", menuName = "Assets/Caracter")]
public class CharacterObject : Item
{
    [Header("Campos básicos")]
    public int numero_Radical;
    public string hanzi;
    public string traduccion;
    public string pinyin;
    public pinyinTono tono;
    public int numero_Trazos;
    public string[] variantes;
    [Space]
    [Tooltip("Sound")]
    public AudioClip audio;
    [Space]
    [Tooltip("Modo casa")]
    public Sprite casa_sprite;
    [Space]
    [Tooltip("Modo historia")]
    public Sprite modo_historia_sprite;
    [Space]
    [Tooltip("Order Pinyin Game Sprite")]
    public Sprite order_pinyin_game_hanzi_sprite;
}
```

Figure 40: Porción de código de la clase `CharacterObject`

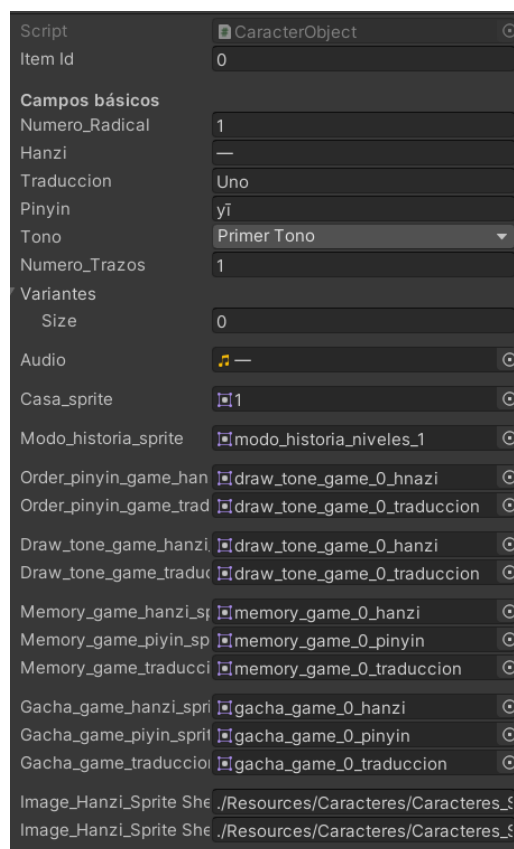


Figure 41: Ejemplo de `Character`

La clase CharacterObject posee además una variable llamada progreso que es en si misma una clase dada su complejidad.

```
[System.Serializable]
public class CharacterProgress
{
    [Header("Identificación de Caracter")]
    public int CharacterID;
    [Header("Campos de Progreso")]
    public bool desbloqueado = false;
    [Range(0,100)]
    public float progreso;
    [Tooltip("Tiempo transcurrido desde la última vez que se repasó el caracter")]
    public float tiempoRepaso;
    [Header("Fallos")]
    public int fallosHanzi;
    public int fallosPinyin;
    public int fallosTraduccion;
    public int fallosSonido;
    public int fallosMinijuegoMemoria;
    public void addProgress(float p)
    {
        progreso += p;
    }
    public void subtractProgress(float p)
    {
        progreso -= p;
    }
    public void unlockCharacter()

```

Figure 42: Porción de código de la clase CharacterProgress

The image shows a Unity Inspector window for a CharacterProgress object. The 'Campos de Progreso' section is expanded, showing the following fields and values:

- Campos Opcionales:**
 - Image_Traduccion_Sprit: [Empty]
 - Image_Traduccion_Sprit: [Empty]
 - Sonido_path: [Empty]
- Campos de Progreso:**
 - Progress: [Expanded]
 - Identificación de Caracter:**
 - Caracter ID: 0
 - Campos de Progreso:**
 - Desbloqueado:
 - Progreso: 0 (Slider)
 - Tiempo Repaso: 0
 - Fallos:**
 - Fallos Hanzi: 0
 - Fallos Pinyin: 0
 - Fallos Traduccion: 0
 - Fallos Sonido: 0
 - Fallos Minijuego Mer: 0

Figure 43: Ejemplo de Progreso

Por otro lado tenemos los tokens, estos objetos son más simples ya que solo poseen un número identificativo, una lista de strings que son los caracteres que componen el grupo y un progreso asignado.

```
[CreateAssetMenu(fileName = "New Sacos Object", menuName = "Assets/Sacos")]
public class SacosObject : ScriptableObject
{
    public int numero_Trazos;
    public string[] caracteres;
    public float progress = 0;
    public void aumentar_grupo(float p){
        progress += p;
    }
    public void disminuir(float p){
        progress -= p;
    }
    public void conertir_progreso_del_caracter_a_grupo(float p, CarácterObject c){
        c.progress.subtractProgress(p);
        progress += p;
    }
}
```

Figure 44: Porción de código de la clase SacosObject

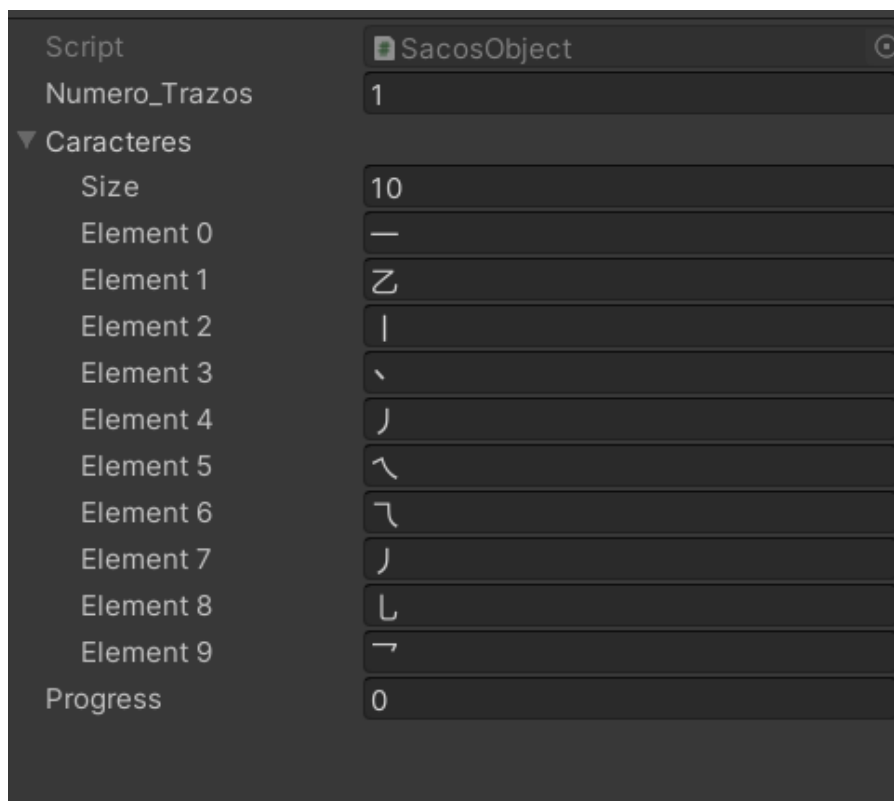


Figure 45: Ejemplo de Saco

Además tenemos el controlador de la base de datos que se encarga de guardar las listas de estos caracteres, tokens y otros datos importantes.

```
[CreateAssetMenu(fileName = "New Database Player", menuName = "Assets/Database/Player Database")]
[System.Serializable]
public class PlayerDatabase : ScriptableObject
{
    public string savePath;

    [SerializeField]
    public Difficulty unlockedDifficulty;
    [SerializeField]
    public Dictionary<Minigames, bool> unlockedMinigames;
    [SerializeField]
    public int unlockedCharacters;
    public void reset()
    {
        unlockedCharacters = 0;
        unlockedDifficulty = Difficulty.easy;
        foreach(Minigames mg in System.Enum.GetValues(typeof(Minigames))) //Recorre el enum de los minij
        {
            unlockedMinigames.Add(mg, false);
        }
    }
    public void unlockNewDifficulty()
    {
        switch (unlockedDifficulty)
        {
            case Difficulty.easy:
                if(unlockedCharacters>50)
                    unlockedDifficulty = Difficulty.normal;
                break;
            case Difficulty.normal:
                if(unlockedCharacters>100)
```

Figure 46: Porción de código de la clase PlayerDatabase

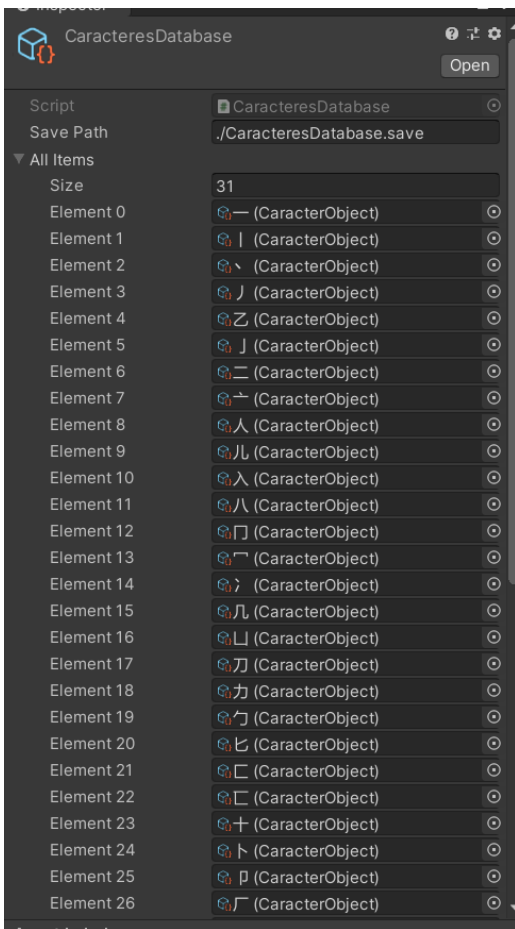


Figure 47: Ejemplo de Base de Datos

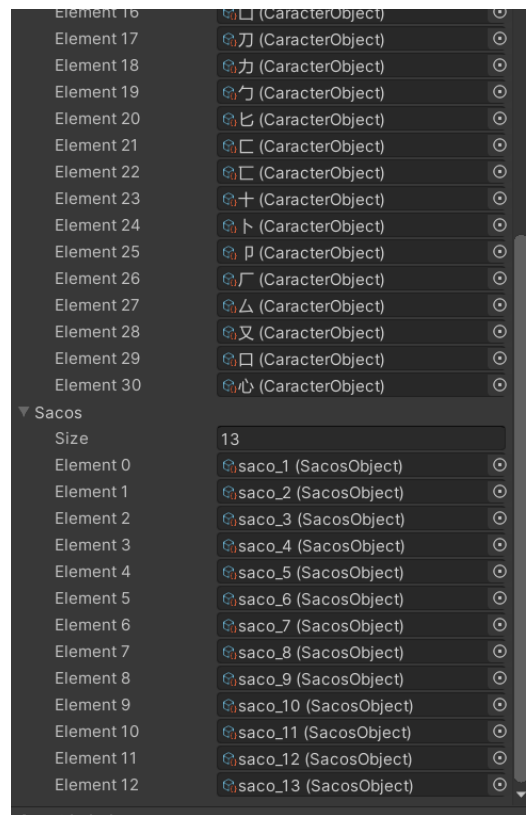


Figure 48: Ejemplo de Base de Datos

6.2 Menús

Los menús es una parte importante del juego ya que es donde el jugador puede comprobar sus estadísticas, seleccionar niveles e interactuar con diversos elementos.

Encargado de gestionar las el resultado de las recompensas aleatorias

```
public class RecompensaController : MonoBehaviour
{
    public Button btn;
    public SacosObject[] sacos;
    public Text txt;
    //Al iniciarse de manera random elegira de 1 a 3 grupos a los que añadirles un % de completado que s
    //Al hacerle click al boton se veran que grupos se han obtenido
    private void Start() {
        btn.interactable = true;
        btn.enabled = true;
    }
    public void obtain_reward(){
        int rnd = Random.Range(1, 4);
        txt.text = "";
        for(int i = 0; i<rnd; i++){
            int rnd_saco = Random.Range(0,sacos.Length);
            int rnd_por = Random.Range(1,5);
            txt.text = "Ne " + sacos[rnd_saco].numero_Trazos + "aumenta " + rnd_por + "%\n";
            sacos[rnd_saco].progress += rnd_por;
        }
        btn.interactable = false;
    }
    public void show(){
        btn.onClick.AddListener(obtain_reward);
    }
}
```

Figure 49: Porción de código de la clase Recompensa controller

Encargado de todas las acciones que suceden en la casa como desbloquear el primer caracter o generar las instancias de los caracteres.

```
using UnityEngine.InputSystem;

public class HouseManager : MonoBehaviour
{
    public CharacterObject[] caracteres;
    public List<GameObject> list_caracteres_object;
    public GameObject prefab;
    public Transform content;

    private void Awake() {
        foreach(CharacterObject character in caracteres){
            GameObject new_character = Instantiate(prefab, content);
            list_caracteres_object.Add(new_character);
            new_character.GetComponent<HouseCaracter>().SetCharacter(character);
        }
    }
    public void unlock_first(){
        caracteres[0].Unlock();
        StateVariablesController.unlocked_caracteres_list.Add(caracteres[0]);
        list_caracteres_object[0].GetComponent<HouseCaracter>().unlocked_reload();
    }
}
```

Figure 50: Porción de código de la clase HouseManager

Encargado de cargar los minijuegos correspondientes en modo practica y comprobar que el minijuego seleccionado esta desbloqueado.

```
GachaGameButton.onClick.AddListener(delegate{ParameterOnClick("GachaGameButton");});
MemoryGameButton = MemoryGame.GetComponent<Button>();
MemoryGameButton.onClick.AddListener(delegate{ParameterOnClick("MemoryGame");});
OrderPinyinGameButton = OrderPinyinGame.GetComponent<Button>();
OrderPinyinGameButton.onClick.AddListener(delegate{ParameterOnClick("OrderPinyinGame");});
ListenGameButton = ListenGame.GetComponent<Button>();
ListenGameButton.onClick.AddListener(delegate{ParameterOnClick("ListenGame");});
}

public void ParameterOnClick(string gameName){
    if((StateVariablesController.show_tutorial_draw_pinyin_game && gameName == "DrawToneGame")||
        (StateVariablesController.show_tutorial_gacha_game && gameName == "GachaGameButton")||
        (StateVariablesController.show_tutorial_memory_game && gameName == "MemoryGame")||
        (StateVariablesController.show_tutorial_order_pinyin_game && gameName == "OrderPinyinGame")||
        (StateVariablesController.show_tutorial_listen_game_game && gameName == "ListenGame")){
        alert.SetActive(true);
        StartCoroutine(DeactiveAlert(3f));
    } else {
        StateVariablesController.selected_minigame = gameName;
        SceneManager.LoadScene("MinijuegoSelectorDificultad");
    }
}

IEnumerator DeactiveAlert(float seconds)
{
    yield return new WaitForSeconds(seconds);
    alert.SetActive(false);
}
}
```

Figure 51: Porción de código de la clase PracticeController

6.3 Inputs

El sistema de input utilizado es la librería New Input System gratuita de Unity, la cual ha permitido personalizar en profundidad que tipos de entradas quiero que el usuario pueda hacer. Se han implementado ciertas acciones como "Arrastrar y soltar", "Hacer click", "Arrastrar"...

Un gestor de inputs generales

```
[DefaultExecutionOrder(-1)]
public class InputManager : Singleton<InputManager>
{
    #region Events
    public delegate void StartPositionPrimaryEvent(Vector2 position, float time);
    public event StartPositionPrimaryEvent OnStartPositionPrimary;
    public delegate void EndPositionPrimaryEvent(Vector2 position, float time);
    public event EndPositionPrimaryEvent OnEndPositionPrimary;
    public delegate void StartTouchPrimaryEvent(Vector2 position, float time);
    public event StartTouchPrimaryEvent OnStartTouchPrimary;
    public delegate void EndTouchPrimaryEvent(Vector2 position, float time);
    public event EndTouchPrimaryEvent OnEndTouchPrimary;
    #endregion
    private TouchControls touchControls;
    private Camera mainCamera;
    private void Awake() {
        touchControls = new TouchControls();
        mainCamera = Camera.main;
    }

    private void OnEnable()
    {
        touchControls.Enable();
    }
    private void OnDisable()
    {
        touchControls.Disable();
    }
    private void Start() {
        touchControls.Touch.PrimaryContact.started += ctx => StartTouchPrimary(ctx);
        touchControls.Touch.PrimaryContact.cancelled += ctx => EndTouchPrimary(ctx);
    }
}
```

Figure 52: Ejemplo de código de input 1

El encargado de gestionar el input de arrastrar el dedo por la pantalla.

```
public class DragFinger : MonoBehaviour
{
    [SerializeField]
    private InputAction mouseClick;
    [SerializeField]
    private float mouseDragPhysicSpeed = 10;
    [SerializeField]
    private float mouseDragSpeed = .05f;

    public GameObject elementToMove;

    private Camera mainCamera;
    private WaitForFixedUpdate waitForFixedUpdate = new WaitForFixedUpdate();
    private Vector3 velocity = Vector3.zero;
    private void Awake(){
        mainCamera = Camera.main;
    }

    private void OnEnable(){
        mouseClick.Enable();
        mouseClick.performed += MousePressed;
    }

    private void OnDisable(){
        mouseClick.performed -= MousePressed;
    }
}
```

Figure 53: Ejemplo de código de input 2

El encargado de gestionar el arrastrar y solar un objeto de la pantalla.


```

}

private void MousePressed(InputAction.CallbackContext context){
    Ray ray = mainCamera.ScreenPointToRay(Mouse.current.position.ReadValue());
    RaycastHit2D hit2D = Physics2D.Raycast(ray); //2D
    if(hit2D.collider != null && (hit2D.collider.gameObject.CompareTag("Draggable") ||
        hit2D.collider.gameObject.layer == LayerMask.NameToLayer("Draggable") ||
        hit2D.collider.gameObject.GetComponent<IDrag>() != null)){
        StartCoroutine(DragUpdate(hit2D.collider.gameObject));
    }
}

private IEnumerator DragUpdate(GameObject clickedObject){
    clickedObject.TryGetComponent<Rigidbody2D>(out var rb);
    clickedObject.TryGetComponent<IDrag>(out var iDragComponent);
    iDragComponent?.onStartDrag();
    float initialDistance = Vector3.Distance(clickedObject.transform.position, mainCamera.transfor
    while(mouseClick.ReadValue<float>() != 0){
        if(clickedObject.GetComponent<Renderer>().enabled){
            Ray ray = mainCamera.ScreenPointToRay(Mouse.current.position.ReadValue());
            if(rb != null){
                Vector3 direction = ray.GetPoint(initialDistance) - clickedObject.transform.positi
                rb.velocity = direction * mouseDragPhysicSpeed;
                yield return waitForFixedUpdate;
            }
            else{
                clickedObject.transform.position = Vector3.SmoothDamp(clickedObject.transform.posi
                ray.GetPoint(initialDistance),ref velocity,mouseDragSp
                yield return null;
            }
        }
    }
}

```

Figure 54: Ejemplo de código de input 3

6.4 Minijuegos

Cinco de los minijuegos planteados originalmente han sido desarrollados, y todos y cada uno de ellos tienen sus peculiaridades.

Todos los minijuegos siguen la misma línea en la que se asigna la dificultad y una lista de caracteres a la que el jugador debe enfrentarse

```

numberOfCharacters = numberOfCharacters * (StateVariablesController.minigame_difficulty * 2);
numberOfItems = numberOfItems * (StateVariablesController.minigame_difficulty * 2);
caracteres = StateVariablesController.current_caracteres_list;
int random_background_index = Random.Range(0, backgroundList.Length);
background.sprite = backgroundList[random_background_index];

for(int i = 0; i<numberOfCharacters;i++){
    bool not_added = true;
    do{
        int randomNum = Random.Range(0, caracteres.Count);
        if (!itemIdentifier.Contains(randomNum)){
            itemIdentifier.Add(randomNum);
            not_added = false;
        }
    }
    while(not_added);
}

```

Figure 55: Código al iniciar un minijuego

6.4.1 Minijuego Dibujar el tono

El minijuego de dibujar el tono debe asignar primero el orden de los vertices correctos.

```

public void set_current_order(CharacterObject hanzi){
    current_points_order = new List<pinyinAreaVertexName>();
    correct_points_order = new List<pinyinAreaVertexName>();
    if(hanzi.tono == pinyinTono.primerTono){
        correct_points_order.Add(pinyinAreaVertexName.A1);
        correct_points_order.Add(pinyinAreaVertexName.A2);
        correct_points_order.Add(pinyinAreaVertexName.A3);
    }
    else if(hanzi.tono == pinyinTono.segundoTono){
        correct_points_order.Add(pinyinAreaVertexName.C1);
        correct_points_order.Add(pinyinAreaVertexName.B2);
        correct_points_order.Add(pinyinAreaVertexName.A3);
    }
    else if(hanzi.tono == pinyinTono.tercerTono){
        correct_points_order.Add(pinyinAreaVertexName.B1);
        correct_points_order.Add(pinyinAreaVertexName.C2);
        correct_points_order.Add(pinyinAreaVertexName.B3);
    }
    else if(hanzi.tono == pinyinTono.cuartoTono){
        correct_points_order.Add(pinyinAreaVertexName.A1);
        correct_points_order.Add(pinyinAreaVertexName.B2);
        correct_points_order.Add(pinyinAreaVertexName.C3);
    }
}

```

Figure 56: Porción de código de la función set current order

6.4.2 Minijuego Gacha

El minijuego de gacha detecta cuando los elementos son arrastrados hacia un tono y comprueba si el tono en el que se encuentran es correcto.

```
void OnCollisionEnter2D(Collision2D collision)
{
    if (collision.gameObject.tag == "gacha_game_tone")
    {
        if ((collision.gameObject.name == "First" && caracter.tono == pinyinTono.primeroTono) ||
            (collision.gameObject.name == "Second" && caracter.tono == pinyinTono.segundoTono) ||
            (collision.gameObject.name == "Third" && caracter.tono == pinyinTono.tercerTono) ||
            (collision.gameObject.name == "Fourth" && caracter.tono == pinyinTono.cuartoTono))
        {
            caracter.progress.addProgress(10);
            sacos[caracter.numero_Trazos -1].progress += 1;
            gameObject.GetComponent<Renderer>().enabled = false;
            GachaGameController.GetComponent<GachaGameController>().AddToDoneList(gameObject);
        }
        else
        {
            caracter.progress.subtractProgress(10);
            sacos[caracter.numero_Trazos -1].progress -= 1;
            GachaGameController.GetComponent<GachaGameController>().securedReward = false;
        }
    }
}
```

Figure 57: Porción de código de la OnCollisionEnter2D

6.4.3 Minijuego Escucha

El minijuego de escucha debe reproducir el sonido cada cierto tiempo mientras el jugador piensa en la respuesta correcta para que así no se olvide de que caracter debe elegir.

```
InvokeRepeating("MakeSound", 2.0f, 4.0f);
}
void MakeSound() {
    audio.Play();
}
```

Figure 58: Código de reproducción de sonido de manera repetida

6.4.4 Minijuego Memoria

El minijuego de memoria debe gestionar que caras estan visibles en todo momento, si estas caras corresponden al mismo caracter...

```
public bool TwoCardsUp()
{
    bool cardsUp = false;
    if(visibleFaces[0] >= 0 && visibleFaces[1] >= 0)
    {
        cardsUp = true;
    }
    return cardsUp;
}

public void AddVisibleFace(int index)
{
    if(visibleFaces[0] == -1)
    {
        visibleFaces[0] = index;
    }
    else if (visibleFaces[1] == -2)
    {
        visibleFaces[1] = index;
    }
}

public void RemoveVisibleFace(int index)
{
    if (visibleFaces[0] == index)
    {
        visibleFaces[0] = -1;
    }
    else if (visibleFaces[1] == index)
    {
        visibleFaces[1] = -2;
    }
}
```

Figure 59: Porción del código del minijuego de memoria

6.4.5 Minijuego Ordenar Pinyin

El minijuego de ordenar el pinyin muestra el orden correcto en caso de que el jugador se equivoque.

```
public void show_correct_pinyin(){
    foreach(GameObject element in letters_go_list)
    {
        if(correct_letter_order.Contains(element.GetComponent<OrderPinyinLetter>().letter_value))
        {
            element.GetComponent<OrderPinyinLetter>().spriteRenderer.sprite = element.GetComponent<O
        }
        else
        {
            element.GetComponent<OrderPinyinLetter>().spriteRenderer.sprite = element.GetComponent<O
        }
    }
    cataacterToShow.GetComponentInChildren<TextMesh>().text = cataacterToShow.GetComponent<ToOrderElem
}
```

Figure 60: Porción del código del minijuego de ordenar pinyin: mostrar patron correcto

7 Resultados

A continuación haré una breve explicación de los resultados respecto a los minijuegos anteriormente implementados y algunas de las complicaciones presentadas de cada uno de estos.

7.1 Minijuego Dibujar Tono

El minijuego de dibujar el tono consta de una tarjeta que representa un caracter sobre el cual hay que adivinar su tono; y un tablero sobre el que dibujar los tonos.

Las tarjetas aparecen de forma aleatoria entre las opciones de una posible lista de caracteres hasta llegar a un determinado número de tarjetas completadas que dependen de la dificultad del juego.

Al dibujar erroneamente un tono, se mostrará el correcto.



Figure 61: Minijuego Dibujar Tono

7.2 Minijuego Gacha

El minijuego de gacha consta de un número de bolas aleatorias que representa un caracter sobre las cuales hay que adivinar su tono; y un cuatro bolas en la parte inferior que representan los tonos hacia los cuales hay que arrastrar las bolas de los caracteres.

Las bolas aparecen de forma aleatoria entre las opciones de una posible lista de caracteres hasta llegar a un determinado número de bolas en pantalla que dependen de la dificultad del juego.

Cada vez que el jugador se equivoque de tono se le restarán puntos pero no se mostrará la respuesta correcta, es decir, puede probar en todas las opciones hasta acertar.



Figure 62: Minijuego Gacha

7.3 Minijuego Escucha

El minijuego de escucha consta de un sonido aleatorio que se repite de forma frecuente; y un cuatro bolas que representan las posibles opciones del jugador para elegir que caracter representa el sonido escuchado.

Los sonidos aparecen de forma aleatoria entre las opciones de una posible lista de caracteres hasta llegar a completar un determinado numero de caracteres que dependen de la dificultad del juego.

No se mostrará la respuesta correcta hasta el final del minijuego, cuando se muestra el feedback. Esto se debe a una falta de tiempo para la implementación de mostrar la respuesta correcta, pero la idea inicial era que tras cada fallo se mostrará la respuesta correcta.



Figure 63: Minijuego Escucha

7.4 Minijuego Memoria

El minijuego de memoria consta de ocho cartas que inicialmente se encuentran boca abajo. El jugador puede seleccionar hasta dos a la vez para intentar emparejarlas. Al fallar el jugador deberá darle la vuelta a una de ellas e intentarlo de nuevo.

Los caracteres que aparecen en parejas son seleccionados de forma aleatoria entre las opciones de una posible lista de caracteres hasta llegar a completar un determinado número de caracteres que dependen de la dificultad del juego.



Figure 64: Minijuego Memoria

7.5 Minijuego Ordenar Pinyin

El minijuego de ordenar el pinyin consta de una tarjeta aleatoria que representa un caracter; y un numero determinado de letras posibles para formar el pinyin. Además, incluye un texto que va formando el pinyin en función de las letras seleccionadas.

Las letras pueden seleccionarse o deseleccionarse en cualquier momento y, al llegar a un número de letras igual que el número de letras que tiene el pinyin del caracter en cuestion se comprueba si es correcto o no. En caso de no serlo se muestra la respuesta correcta..

Las tarjetas aparecen de forma aleatoria entre las opciones de una posible lista de caracteres hasta llegar a completar un determinado numero de caracteres que dependen de la dificultad del juego. Las letras aparecen de forma aleatoria dentro de un listado que incluye todas las consonantes y todas las vocales con los 4 tonos.



Figure 65: Minijuego Ordenar Pinyin

Al comenzar un minijuego, la primera vez que el jugador se encuentre con este reto tendrá una pequeña explicación sobre como jugar. Al finalizar el listado de caracteres, se mostrará la pantalla de feedback.

Para ver más detalles del funcionamiento ver el video con la demo que se encuentra en el siguiente link:

<https://drive.google.com/file/d/1n3o87B7xpbP6ugmCoT4ZLqBQZ4dCDC1b/view?usp=sharing>

8 Conclusiones

A modo de conclusión, el videojuego que he planteado al inicio del desarrollo era mucho más grande de lo que en un principio creía, por lo tanto llevará más tiempo de trabajo del planeado ya que hay muchas mecánicas por implementar, mucho arte que crear e integrar.

Por otro lado el resultado de la demo es positivo ya que esta posee unas sólidas bases que hacen que la ampliación del juego sea sencilla, tanto por la parte de introducción de nuevos caracteres y niveles como por la parte de creación de nuevos y diversos minijuegos entretenidos.

Una de las mayores complicaciones a la hora de desarrollar el juego ha sido la poca compatibilidad de unity con el idioma Chino, lo cual ha dificultado el desarrollo de algunas partes del videojuego como cuando algunos caracteres no aparecen con ciertas fuentes cuando son mostradas por pantalla.

Finalmente creo, que estoy satisfecha con el trabajo realizado y con los resultados obtenidos.

9 Trabajo futuro

Para el trabajo futuro tengo planeadas varias ramas, la rama artística y cultural; la rama de aprendizaje y la rama de entretenimiento. Para la rama artística y cultural me gustaría añadir más variedad en la estética, haciendo que los escenarios varíen en función del avance en las habilidades del jugador. Es decir, a medida que el jugador complete niveles de historia y domine caracteres la estética cambiará. Este cambio en la estética se basará en un avance por las diversas dinastías de los más de 5000 años de historia de China mostrando estos cambios en el entorno y los personajes. Tanto los fondos, el vestuario y los diversos objetos avanzarán junto con las dinastías. Además, añadir una narrativa que profundice más en la cultura de China. Para la rama de aprendizaje me gustaría añadir más caracteres y palabras a aprender. Estas palabras estarían organizadas en grupos de palabras a los cuales el jugador puede acceder en sub-mundos dentro del mismo mundo (en el modo historia), es decir, por ejemplo: en el mundo lineal se encuentran los radicales y, al llegar a ciertos radicales el jugador desbloqueará la posibilidad de elegir un grupo de caracteres o palabras para entrar en un sub-mundo. Algunos de estos grupos podrían ser los colores, las profesiones, los verbos, los números... Para la rama de entretenimiento me gustaría añadir más variedad de minijuegos para que el jugador no experimente nunca aburrimiento y pueda aprender variando los minijuegos. Además de más minijuegos, me gustaría añadir más opciones en el mundo como pueden ser logros, una casa en la que el jugador puede poner los trofeos obtenidos, un sistema de obtención de monedas en función del avance por el mundo y la mejora de habilidades, compra de objetos decorativos para la casa con las monedas obtenidas...

Al ser una demo, no se ha podido implementar todo lo deseado y algunas partes del apartado artístico del juego entrarían como trabajo a futuro: el progreso de los caracteres, los botones entre partes del mundo, el feedback al acabar un minijuego, los minijuegos de detectar errores y de escucha, opciones de guardar y cargar la partida en local...

Para adaptar que el juego sea más accesible, en el futuro se implementarán las opciones de cambio de tamaño de fuente y de lectura del texto en sonido.

10 Bibliografía

- Pinyin
<https://es.wikipedia.org/wiki/Pinyin#: :text=El%20h%C3%A0ny%C7%94%20p%C4%ABny%C4%>
- Radical
[https://es.wikipedia.org/wiki/Radical_\(sinograma\)#: :text=El%20t%C3%A9rmino%20radical%20\(e](https://es.wikipedia.org/wiki/Radical_(sinograma)#: :text=El%20t%C3%A9rmino%20radical%20(e)
- Trazos
<https://www.saporedicina.com/es/escritura-china-reglas-trazos/>
- Hanzi
<https://es.wikipedia.org/wiki/Sinograma>
- Tonos
https://es.wikibooks.org/wiki/Chino/Pronunciaci%C3%B3n/Los_tonos
- Inspiración artística
https://br.pinterest.com/hanzi11831/_saved/
- Lista de radicales chinos
<https://www.hanzibox.com/lista-de-radicales-chinos/>
- Serialize a Class in Unity
<https://www.youtube.com/watch?v=MVmWbmN9Iv0>
- Json y PlayerPrefs
<https://www.youtube.com/watch?v=sIPOVSGeNHE>
- Sistema de inventario y guardado de inventario
<https://www.youtube.com/channel/UCUPvKyKhh33gn0VatEjCrcQ>
- Inspiración artística
<https://merchanthistoryprojectnicholascavanaugh.weebly.com/role-in-ancient-chinese-society.html>
- Hablantes de chino
<https://es.statista.com/estadisticas/635631/los-idiomas-mas-hablados-en-el-mundo/>
- Examen imperial Chino
https://es.wikipedia.org/wiki/Sistema_de_examen_imperial_chino
- Arte chino antiguo
https://en.wikipedia.org/wiki/Chinese_art#: :text=In%20ancient%20China%2C%20painting%20an
- Radicales agrupados por n^o de trazos
<https://www.mdbg.net/chinese/dictionary?page=radicals>
- Conversor de texto a sonido
<https://soundoftext.com/>

- N^o de trazos de los hanzi
<http://hanzidb.org/character-list/by-stroke-count>
- Musica
<https://www.chosic.com/free-music/chinese/>
- Musica
<https://www.fesliyanstudios.com/royalty-free-music/downloads-c/chinese-music/61>