**Programación sobre redes**

**Trabajo Práctico VideoJuego**

**Profesor: Lic. Leonardo Cancino.**

**Alumnos: Osman Guaqui, Ariadna Rivero, Zamira Nozralah, Alex Quispe Ponce**

**Año y División: 6**° **3ª**

**Ciclo Lectivo: 2023**

**Propuesta**

El video juego “Martin 1976” se trata de un juego didáctico y educativo, por lo que nuestro principal público serán el profesorado y el alumnado del ciclo básico de secundaria. El objetivo de “Martin 1976” es el de concientizar sobre una de las peores etapas (Dictadura militar 1976) que ha tenido la Argentina que si bien se habla nunca se profundiza como se debería. Además “Martín 1976”, es un videojuego desarrollado para alumnos por alumnos, por lo que comprendemos que la metodología de enseñanza a veces se torna pesada, por esto queremos dar paso a esta nueva herramienta de enseñanza a las aulas.

“Martin 1976”, es un juego en 2D de género JRPG con elementos de trivia. El juego pone al jugador en la piel de Martín, que será el personaje principal que se aventura en la etapa de la dictadura militar de 1976 mostrando los sucesos más importantes de esta etapa, por lo que el jugador tiene que explorar el mapa interactuando con otros personajes y objetos, que brindaran información o darán datos a tener en cuenta de la historia y del juego.

El jugador tiene como desafío después de cada “episodio” el contestar una trivia con preguntas de diversos temas tanto referidos al juego como no. A su vez estas trivias tendrán un sistema de recompensas y penalizaciones que dificulten el avance del jugador. Por cada episodio superado por el jugador, se dará una sinopsis de todo el contexto histórico del episodio, como también una introducción hacia el siguiente.

En cuanto a la duración del juego se estima que será de 45 minutos con el adicional de las trivias, ya que el videojuego sirve para adentrarse en el tema y que se haga hincapié en los sucesos contados en el juego.

***Implementación***

* **Conceptualización:** En este paso se inició con una lluvia de ideas del equipo de trabajo, en donde lo que resaltó más fue la idea inicial de “Martin 1976”, que en un principio fue llamado “Naval Trivia”. La idea era desarrollar un juego narrativo con elementos de trivia que narra sucesos históricos mundiales, sin embargo ante la gran inmensidad de los sucesos históricos nos dispusimos a hacerlo de una etapa que consideramos importante recordar por todo lo transcurrido y el impacto que generó hasta la actualidad, osea la dictadura militar ocurrida en 1976.
* **Planificación:** La planificación se hizo una vez se tuvo claro el concepto del juego, por lo que recurrimos a distintas herramientas para llevar a cabo el desarrollo del juego, pero no sin antes definir preguntas básicas, para tener cimientos sobre los cuales empezar como tal con la planificación para esta etapa usamos diagramas y herramientas tales como: Ganttproject, diagrama de clases, casos de uso, trello, etc.

Las preguntas básicas que hicimos en concreto para el desarrollo de “Martin 1976” fueron: ¿Qué género será el videojuego? ¿Cuáles son las dimensiones? ¿De qué tratará? ¿Tenemos las habilidades para hacerlo? ¿Qué público tendrá? ¿Cómo lo haremos? ¿Qué herramientas usaremos? ¿Por medio de qué plataforma lo desarrollaremos? ¿Cuánto tiempo tenemos?

* **Diseño:** Esta etapa fue la que nos encaminó al inicio del desarrollo del video juego como tal, porque es la etapa en donde pasamos “en limpio” todas las ideas y respuestas que tuvimos de las preguntas hechas en la etapa anterior, de manera que nos dio el primer avance en cuanto a los diseños de pantalla, el diagrama de clase, casos de uso, prototipos de personajes y escenarios, etc. Esta etapa es en la cual verificamos con que estamos trabajando en sí, ya que da paso a la producción del video juego.
* **Producción:** La producción es la etapa más duradera y desafiante del proyecto, ya que, es en donde empezamos a crear el videojuego en sí. El equipo de programadores junto al de diseñadores serán el foco principal en esta etapa, ya que, serán los encargados que el video juego se desarrolle tanto en su “Front end” (Diseñadores) como en su “Back end” (Programadores)
* **Pruebas:** Una vez terminó la etapa de producción llegó a su meta, la etapa de pruebas se encarga de detectar y corregir los posibles fallos que puedan haber en el juego, esta etapa será llevada a cabo por el mismo grupo de trabajo del proyecto.

**Guía Técnica**

Lenguaje que utilizaremos: Java

Se usará Java como lenguaje para programar el juego debido a que es un lenguaje en el cuál el equipo de desarrollo se siente más seguro, y cómodo, ya que es el lenguaje que aprendimos más a fondo de todos los que vimos y tenemos más conocimientos por ende sabemos manejarnos de una mejor manera debido a que ya tenemos una base en la cual apoyarnos. Mientras que Python al ser un lenguaje desconocido por el equipo de trabajo se torna más pesado el trabajo y nos trae más dificultades que beneficios, ya que también juega en contra el tiempo que tiene que cumplir el equipo de desarrollo para la producción del video juego en sí, abarcando documentación y producción del mismo. Al principio del proyecto cuando definimos en aspectos técnicos como íbamos a trabajar elegimos Python, pero viendo que la lógica de crear un videojuego en java es prácticamente la misma (sprites, eventos, detección de colisiones, método paint, actualizar la pantalla por cada evento que se produzca..) optamos como mejor opción inclinarnos por java en vista del tiempo que tenemos.Por otro lado java tiene un proceso más rápido que Python a la hora de llamar funciones, java tiene menos abstracción y al llamar un método solo tiene que recorrer en la tabla de métodos virtuales hasta localizar el que queramos buscar. Tampoco nos será necesario instalar frameworks externos eligiendo Java como lenguaje. Además, el juego nos va a quedar mejor estructurado en el código, lo que ayuda a que haya un mayor entendimiento del producto para los programadores.

IDE: Eclipse

Framework: No vamos a utilizar librerías externas ya que nos vamos a manejar simplemente con las herramientas que ya nos proporciona el entorno de desarrollo(Eclipse).

Una clase **Sprites** en Java es una clase base diseñada para el **movimiento de imágenes**. Los Sprites son rectángulos que representan los objetos móviles o fijos de un juego. Esos rectángulos pueden ser cualquier elemento de un juego, como en nuestro juego serán el jugador y los personajes NPC, los obstáculos, las herramientas, entre otros objetos.

* **Estructura que tendrá el desarrollo del juego :**

1. **Se crea y configura el bucle del juego**, con los elementos que va a tener nuestro juego.

2. **Se comprueban los eventos**: Dentro de un bucle, comprobamos los posibles eventos que se han producido, por ejemplo, hemos pulsado, o hemos pulsado la tecla ‘e’ para interactuar con un objeto del escenario.

3. **Se actualiza la pantalla**: Según la lógica del juego o de algún evento que haya sucedido, se modifican los elementos (por ej: el NPC dice su diálogo, o se abre una puerta) y se vuelve a dibujar en la pantalla.

* **Animación del juego**

La animación del personaje y de los npcs, es en realidad una serie de imágenes estáticas que parece que están en movimiento por la cantidad de veces que se actualiza la pantalla por segundo(FPS). Como las imágenes se actualizan muy rápidamente da la sensación de que el personaje se está moviendo. Si nuestro juego funciona a 60 FPS esto se plasmará como una serie de 60 imágenes en pantalla por segundo.

Para crear esta animación necesitamos crear nuestro propio reloj de tiempo en el juego, y en JAVA podemos hacerlo creando un hilo utilizando la clase Thread. La clase Thread nos es útil para cuando necesitamos crear procesos que se repiten, como por ejemplo: actualizar la pantalla 60 veces por segundo, “pintar” la pantalla cada vez que se actualice con la nueva imágen, o iniciar una partida. Un hilo se puede iniciar y una vez iniciado el programa se ejecutará hasta que el hilo se detenga o termine su flujo de instrucciones(si el jugador por ejemplo inicia una partida, mientras siga jugando este hilo va a seguir corriendo, hasta que el jugador termine la partida o salga del juego).

***Patrón de diseño***

**MVC(Modelo Vista Controlador)** → Patrón arquitectónico de software

El **modelo** se encarga de definir las entidades básicas del videojuego, en nuestro caso serían el jugador y los npc con los que interactúa. También los accesorios que pueden utilizar estas entidades o en nuestro caso las que estarán esparcidas por el mapa(interactivas).

El **controlador** por su parte se encarga de manejar el modelo y recibir eventos del mismo, también define cómo se gana y como se pierde en el juego.

La **vista** es la interfaz con la cuál el usuario interactúa directamente en el juego, es decir todo lo que verá el usuario al jugar(las ventanas). También se encarga de manejar el controlador y de escuchar eventos del controlador, como saber cuando el juego termina.

En Java cada uno de estos componentes representarán un package del project. Con sus respectivas clases dentro.

**Clases Identificadas:** Jugador, NPC, objeto, escenario.

***Propuesta de producción***

Se usarán las siguientes herramientas para la organización del equipo y desarrollo del proyecto:

* El ciclo de desarrollo será una combinación entre Secuencial y Solapado, es decir habrá tareas que son dependientes de sus antecesoras para ser comenzadas, y habrá otras que no necesitan tener ninguna antecesora.
* A su vez se usará también los roles scrum, ya que, concuerdan con los roles ya definidos para el proyecto.
* Se usarán los sprints para la organización de tareas y de entregas de manera que se pueda lograr un desarrollo incremental óptimo
* Las ceremonias Scrum serán una de las herramientas cruciales, ya que, serán las reuniones del equipo para planificar el proyecto y será donde se hará frente a los inconvenientes que puedan surgir, por lo que se utilizara el Scrum diario, Scrum de Scrum, planificación del Sprint y revisión del Sprint
* El lenguaje de programación será Java
* Los diagramas y documentos en sí serán una de las principales herramientas para el desarrollo del proyecto, por lo que se contará con los siguientes: Diagrama de flujo - Diagrama de clases - Gantt - DER - Diagrama de casos de uso - Documento de diseño - entre otros.

***Procesos viables para el desarrollo del juego:***

1. Documentación de la trama: Los documentadores se encargarán de plasmar en un archivo de texto un resumen de los hechos ocurridos durante la segunda dictadura militar que tuvo lugar en la Argentina. Con el fin de poder abordar con mayor facilidad todo el trayecto del videojuego respecto al modo historia y los bocetos de los contenidos que tendrán las distintas pantallas.
2. Información del producto: Los documentadores tendrán la tarea de investigar cuáles son los componentes que debe tener un videojuego 2D creado en Java; cuál es el paradigma, manejar eventos, como crear un objeto y hacer que se mueva, una ventana vacía y agregarle componentes..
3. Capacitación 1: Los programadores consultarán distintos enlaces web y vídeos/tutoriales de aprendizaje básicos para programar en Java para así familiarizarse con el lenguaje a trabajar en el proyecto siguiendo la playlist a continuación y haciendo ejercicios prácticos.  
   Playlist: <https://www.youtube.com/watch?v=qa6GA5p9nQ0&list=PLN9W6BC54TJJr3erMptodGOQFX7gWfKTM>
   1. Capacitación 2: Una vez ya adquiridos los conocimientos generales para introducirse en Java y más específicamente en el desarrollo de videojuegos
4. Bocetos de las pantallas: Antes de diseñar digitalmente las pantallas finales del videojuego, es conveniente tener un bosquejo general de las más importantes o las que son diferentes, como pueden ser; el menú con sus opciones, trivias de los niveles, configuración del juego.
5. Casos de uso: Generar diagramas de c.u para poder entender con mayor facilidad cómo se comportará la interfaz con el usuario.
6. Metodología de trabajo: Establecer una metodología eficaz y productiva para el equipo de trabajo. De esta forma se logrará mejor calidad de trabajo y una organización más efectiva.
7. Guión: Escribir todos los diálogos y cuadros de texto de los personajes

| **CAMPO** | **DESCRIPCIÓN** |
| --- | --- |
| **CONCEPTO** | |
| Título | Naval Trivia |
| Estudio/Diseñadores | Osman Guaqui, Ariadna Rivero, Zamira Nozralah, Alex Quispe Ponce |
| Género | Aventura - JRPG ( Narrativo con elementos de trivia) |
| Plataforma | PC |
| Versión | 1.0 |
| Sinopsis de Jugabilidad y Contenido | El jugador recorre en vista de [Personaje principal] por el último golpe de estado de argentina, encontrándose con distintas dificultades en el camino que desafiaron su conocimiento y la atención a los detalles que puede haber en el juego. |
| Categoría | CodeCombat - Chemcaper. La similitud que hay con estos juegos son el género del videojuego y el objetivo con el que se diseñaron, es decir todos son para el aprendizaje así sean en distintas materias. La diferencia de naval trivia con estos videojuegos radica en la mecánica del juego y los objetivos secundarios que tiene. |
| Licencia | No Aplica |
| Mecánica | La mecánica principal del juego se basará en la aventura, por lo que el personaje se irá moviendo en los distintos escenarios. Cada vez que se cambia de un escenario a otro se da una sinopsis de sucesos del golpe de estado. A su vez los escenarios contarán con distintos NPC con los que se pueda interactuar, para que el jugador conozca datos adicionales y que puedan servir o no para la trivia que se realice.  Cada cambio de escenario contará con una trivia la cual definirá si el jugador es “apto” para el siguiente nivel o no. A su vez se incorpora un sistema de recompensas que serán mostrados en pantalla en forma de “favor” y un sistema de penalizaciones. |
| Tecnología | Software:  Hardware: |
| Público | El público al cual está dirigido Naval Trivia son los estudiantes de ciclo básico de secundaria, sin embargo también puede ser para estudiantes de primaria. |
| **HISTORIAL DE VERSIONES** | |
| 1.0 | |
| **VISIÓN GENERAL DEL JUEGO** | |
| Naval Trivia, es un proyecto desarrollado por estudiantes, para estudiantes, por lo cual entendemos las dificultades de aprendizaje por falta de entusiasmo, por eso traemos un videojuego que brinda entretenimiento, es didáctico y de auto-aprendizaje.  Este proyecto es un desafío e impulso a los profesores y estudiantes para incorporar algo fuera de lo común a las aulas. También queremos incentivar al auto-aprendizaje, porque si bien la tecnología avanza constantemente, sentimos que la educación cada vez se centra más en algo no práctico, para el alumnado.  El proyecto servirá en las aulas como herramienta de introducción a la hora de hablar de un momento histórico en la argentina, el golpe de estado de 1976, por lo que se narran fragmentos, por lo cual el estudiante tendrá que prestar especial atención a los escenarios y las momento narrativos, para que pueda responder algunas de las preguntas en la trivia, a su vez el estudiante podrá interactuar con distintos elementos, que aportaran datos adicionales a la trama del juego o simplemente entretenidos. | |
| **MECÁNICA DEL JUEGO** | |
| El jugador podrá moverse libremente en los escenarios diseñados. Lo hará mediante el teclado  El jugador podrá interactuar con los NPC y elementos específicos. Se tendrá que acercar y presionar un botón en específico para iniciar la interacción.  El jugador podrá omitir ciertas cinematografías. Presionando Enter.  El jugador podrá retroceder diálogos de texto en la cinematografía. Presionando “retroceder”.  El jugador podrá elegir una opción a las preguntas de la trivia presionando click derecho encima de esta opción.  El jugador podrá elegir un “beneficio” en la trivia presionando click derecho en el icono que se muestre en pantalla.  El jugador podrá recoger “favores”. | |
| Cámara | La cámara que se usará para Naval Trivia será “TOP-DOWN”, es decir que se mostrará desde un plano aéreo. El juego será en 2D con pixelart. |
| Periféricos | Teclado - Mouse - Auriculares o parlantes. |
| Controles | La movilidad vertical y horizontal serán con las teclas “W-A-S-D” correspondiendo “W” Arriba, “S” Abajo, “A” Izquierda " y"D " derecha. La interacción será mediante la tecla “E”.  La selección tanto de opciones como de beneficios a la hora de la trivia será con el “Click Derecho”,  La tecla enter servirá para omitir diálogos.  La tecla “Space” servirá para avanzar al próximo diálogo.  La tecla “retroceder” servirá para volver hacia atrás en los cuadros de texto de la cinematografía o en los diálogos  La tecla “Esc” servirá para abrir el Menú |
| Puntaje | Explicar de qué manera el juego se mantiene al tanto de los logros del jugador. Incluir también si existe una tabla de puntajes que compare los mismos entre los jugadores, ya sea de manera local o en línea.  La comparación se realizará de manera local, con el uso de una BD |
| Guardar/Cargar | No aplica |
| **ESTADOS DEL JUEGO** | |
| Un estado del juego se refiere al lugar en donde se encuentra el jugador durante el juego, es decir, si el jugador está en el Menú Principal, está jugando un Juego Multijugador, está en el Menú de Pausa, etc. Los diagramas deben representar visualmente las relaciones entre los estados, si del Menú Principal se puede ir al Menú de Opciones, ¿Cómo lo hace? ¿Qué se ejecuta? ¿Qué interfaz muestra?  Estado Menú Principal: Este estado será el primero de todos, ya que, es el que se mostrará por pantalla dada la ejecución del juego, de manera que mostrará los botones del juego que serán “jugar”, “Opciones”, “Salir” y “Créditos”.  Estado “Juego”: Este estado se dará cuando el jugador esté jugando al video juego, por lo cual se mostrará al jugador, el escenario con los NPC y objetos, y un HUD que muestra el episodio en el que se encuentra, cuantos “favores” tiene y si posee alguna penalización.  Estado “Opciones”: Será el estado en el que el jugador ingresa a “Configuración” desde el “Menú Principal” o desde el “Menú de Pausa”: En la configuración podrá subir o bajar el volumen y consultar los controles del juego.  Estado “Menu de Pausa”: El estado “Menu de pausa” será el menú que aparece cuando el jugador estando en juego presione “esc”: En este se mostrarán los botones “Continuar”, “Configuración” y “Salir”.  Estado “Game over”: El estado “Game Over” prácticamente será cuando el jugador pierde todas sus vidas, por lo cual se mostrará un apartado con los botones de “Jugar de nuevo” y “Salir”. | |
| **INTERFACES** | |
| El video juego será en 2D, con una cámara estilo “TOP-DOWN” el arte que se usará para el diseño del juego será en pixelart. La paleta de colores… | |

| Nombre de la Pantalla | Menú principal |
| --- | --- |
| Descripción de la Pantalla | Será la pantalla principal y la primera que verá el usuario cada que ejecute el juego. En esta pantalla se verán las opciones tales como: Nuevo Juego - Configuración - Cargar Juego y Salir |
| Estados del Juego |  |
| Imagen | - |
| Nombre de la Pantalla | Configuración |
| Descripción de la Pantalla | En esta pantalla se verá el volumen, acompañado de un botón de “Silenciar”. La opción de poner pantalla completa con un recuadro de “SI - NO”. |
| Estados del Juego | - |
| Imagen | - |
| Nombre de la pantalla | Créditos |
| Descripción de la pantalla | En esta pantalla se mostrarán los créditos, es decir las menciones correspondientes a cada uno de los aspectos de la producción. |
| Estados del juego | - |
| Imágenes | - |
| Nombre de la Pantalla | Menú de pausa |
| Descripción de la Pantalla | Este menú será dentro del juego, por lo que contarán con las siguiente opciones: Volver al menú principal, configuración y Salir del juego. |
| Estados del Juego | - |
| Imagen | - |
| **NIVELES** | |
| Los juegos comúnmente se dividen en niveles o en mapas secuenciales dentro de los cuales se debe cumplir con ciertos objetivos para progresar en el juego. Existen juegos en los cuales los niveles solo cambian a razón de la dificultad y los objetivos siguen siendo los mismos, de igual manera se deben describir esos cambios en esta sección. | |
| Título del Nivel | “El principio del fin democratico…” |
| Encuentro | Introducción |
| Descripción | En este encuentro, se narra el inicio de la dictadura militar, es decir, el minuto a minuto de cómo se dio el golpe de estado. |
| Objetivos | ¿Qué debe de hacer el jugador para terminar el nivel? Este campo también debe incluir si el jugador tiene que resolver ciertos acertijos o derrotar a cierto enemigo para progresar. No aplica. |
| Progreso | Una vez termina la narrativa, se pone en marcha la aventura, osea el “episodio 1” |
| Enemigos | No aplica. |
| Items | No aplica. |
| Personajes | No aplica. |
| Música y Efectos de Sonido | Describir la música de este nivel al igual que los efectos de sonido de ambiente que contiene. |
| Referencias de BGM y SFX | Escribir todas las referencias que apliquen con respecto a la música de fondo y efectos de sonido descritos en la sección de Música y Sonidos. |
| **PROGRESO DEL JUEGO** | |
| Enlistar de manera secuencial o por medio de un diagrama de flujo los eventos o niveles que el jugador debe de pasar para progresar en el juego. Existen juegos que tienen distintos modos de juego, en ese caso se requieren varias listas y/o diagramas.  Introducción - Episodio 1 - Episodio 2 - Episodio 3 y Episodio 4.  . | |
| **PERSONAJES** | |
|  | |
| Nombre del Personaje | Martin |
| Descripción | Martin viste como un adolescente de la época, hoy viste con una remera naranja y un jean azul. |
| Imagen |  |
| Concepto | Martin es una persona muy calmada, no es inteligente, en la escuela es un estudiante promedio, es una persona valiente ya que buscará salir de cada uno de los problemas que se le interponga en su vida. |
| Encuentro | Episodio 1. |
| Habilidades | Uso de favores |
| Armas | No aplica |
| Items | No aplica. |
| Personaje No-Jugable | No aplica |
| Nombre del Personaje | “Milico” |
| Descripción | Milico es un humano, un militar que siempre está vestido como tal, no se cambia la vestimenta ni para dormir. |
| Imagen |  |
| Concepto | Milico es una persona nacida para el mal, es muy fuerte ya que estuvo entrenando toda su vida para ser un militar, el no te ayudará, sino que te ocasionará problemas en el transcurso de la historia. |
| Encuentro | Episodio 1. |
| Habilidades | Uso de favores |
| Armas | No aplica |
| Items | No aplica. |
| Personaje No-Jugable | No aplica |
| Nombre del Personaje | “Laucha” |
| Descripción | Describir detalladamente el físico del personaje, si es humano o extraterrestre, su vestimenta, etc.  Lauti es un humano, la vestimenta de Lauti viene a revolucionar la moda, viste unos pantalones jean celestes, y un sueter negro, junto a la barba y al corte le dan una gran personalidad y sobre todo facha. |
| Imagen |  |
| Concepto | Laucha es un egresado de la escuela en donde estudia Martin, fue abanderado, por ende este tiene una inteligencia altamente superior al resto, será de gran ayuda para resolver algunos de los misterios que le deparen al protagonista de esta historia. |
| Encuentro | Episodio 1. |
| Habilidades | Uso de favores |
| Armas | No aplica |
| Items | No aplica. |
| Personaje No-Jugable | No aplica |
| Nombre del Personaje | “Blito” |
| Descripción | Describir detalladamente el físico del personaje, si es humano o extraterrestre, su vestimenta, etc.  Pablito, un humano, suele vestir ropa deportiva, en este caso hoy traera una remera rosada y un pantalon deportivo negro, con unos zapatos deportivos blancos. |
| Imagen |  |
| Concepto | Blito, es un adulto, profesor de la escuela que por muchos es considerado un profe muy copado, suele estar tomando mate, cualquier duda que tengas, Pablito te dará la ayuda. |
| Encuentro | Episodio 1. |
| Habilidades | No aplica |
| Armas | No aplica |
| Items | No aplica. |
| Personaje No-Jugable | No aplica |
| **ENEMIGOS** | |
| NO APLICA | |
| Nombre | El nombre del enemigo. |
| Descripción | Describir detalladamente el físico del enemigo así como también su comportamiento. |
| Encuentro | ¿Cuándo aparece este enemigo en el juego? |
| Imagen | Fotografía o dibujo conceptual del enemigo. |
| Habilidades | Enlistar las habilidades del enemigo. |
| Armas | Enlistar las armas del enemigo. |
| Items | Enlistar los objetos del enemigo. |
| **HABILIDADES** | |
| Los personajes y los enemigos llegan a tener ciertas habilidades fuera de las acciones comunes, en esta sección se describen cada una de ellas. | |
| **ARMAS** | |
| En esta sección se describen las armas que aparecerán en el juego. | |
| **ITEMS** | |
| Todos los objetos especiales que ayudan al jugador a realizar los objetivos y progresar en el juego se mencionan aquí. | |
| **GUIÓN** | |
| En esta sección se incluyen todos los diálogos del juego. Estos pueden ser muy variantes o inexistentes dependiendo de la naturaleza del juego. El guión debe incluir encabezados, nombres, diálogo, acción y transiciones. | |
| **LOGROS** | |
| Describir los varios logros o hitos que el jugador obtiene mientras progresa en el juego. Estos pueden otorgar medallas, personajes secretos o puntos extra. | |
| **CÓDIGOS SECRETOS** | |
| Describir los códigos secretos que el jugador puede ingresar, lo que hacen y cómo son ingresados. | |
| **MÚSICA Y SONIDOS** | |
|  | |
| **IMÁGENES DE CONCEPTO** | |
| Todas las imágenes que muestren algún posible nivel, personaje, objeto, etc., deben ser incluidas en esta sección y deben estar enumeradas y con título. | |
| **MIEMBROS DEL EQUIPO** | |
| Información de las personas que trabajarán en el proyecto, incluye su nombre, el rol o roles que desempeñan y medios por los cuales se les puede contactar. | |
| **DETALLES DE PRODUCCIÓN** | |
| Antes de entrar a la etapa de Producción, se definen en el documento algunos detalles del proyecto. | |
| Fecha de Inicio | ¿Cuándo empieza la etapa de Producción del proyecto? |
| Fecha de Terminación | ¿Cuándo termina la etapa de Producción del proyecto? |
| Presupuesto | Una estimación aproximada del presupuesto del juego. |

Casos de uso:

| Caso de uso | Iniciar una nueva partida. |
| --- | --- |
| Actores | Jugador. |
| Tipo | Principal. |
| Precondición | Tener instalado el juego. |
| Postcondición | No aplica. |

| Propósito | Comenzar a jugar. |
| --- | --- |

| Resumen | El jugador empezará la partida tras apretar el botón. |
| --- | --- |

| Curso normal | |
| --- | --- |
| 1 | El jugador debe hacer click en el botón de “Nueva Partida”. |
| 2 | El sistema cargará una partida nueva. |
| 3 | El jugador ingresará a una nueva partida. |

| Caso de uso | Salir del juego. |
| --- | --- |
| Actores | Jugador. |
| Tipo | Principal. |
| Precondición | Entrar al juego. |
| Postcondición | No aplica |

| Propósito | Cerrar el programa |
| --- | --- |

| Resumen | El programa se cerrará tras haber hecho click en el botón de “Salir”. |
| --- | --- |

| Curso normal | |
| --- | --- |
| 1 | El jugador debe hacer click en el botón de “Salir”. |
| 2 | El jugador saldrá del programa. |

| Caso de uso | Uso de favor (2 opciones menos) |
| --- | --- |
| Actores | Jugador |
| Tipo | Principal |
| Precondición | El jugador debe poseer “favores” |
| Postcondición | Reducir los “favores” |

| Propósito | Reduce la cantidad de opciones de respuesta de la trivia, es decir, en vez de 4 opciones se eliminan 2. |
| --- | --- |

| Resumen | El jugador presiona en el icono de “Reducción de opciones”, se verificará que tenga los favores necesario y a continuación se eliminaran 2 opciones de la trivia. |
| --- | --- |

| Curso normal | |
| --- | --- |
| 1 | El jugador presiona el icono |
| 2 | El sistema verifica si tiene favores |
| 3 | El sistema reduce los favores |
| 4 | El sistema elimina 2 opciones de la trivia. |

| Caso de uso | Uso de favor (Opción correcta) |
| --- | --- |
| Actores | Jugador |
| Tipo | Principal |
| Precondición | El jugador debe poseer “favores” |
| Postcondición | Reducir los “favores” y elegir la opción correcta |

| Propósito | Este favor hará que el jugador automáticamente responda correctamente la pregunta. |
| --- | --- |

| Resumen | El jugador presiona en el icono de “Opción correcta”, se verificará que tenga los favores necesarios y a continuación se responde de manera correcta la pregunta. |
| --- | --- |

| Curso normal | |
| --- | --- |
| 1 | El jugador presiona el icono |
| 2 | El sistema verifica si tiene favores |
| 3 | El sistema reduce los favores |
| 4 | El sistema responde correctamente la pregunta. |