



UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERIA

Escuela Profesional de Ingeniería de Sistemas

Proyecto “*RenatitoBot*”

Curso: Diseño y Creación de Videojuegos

Docente: Ing. Patrick José Cuadros Quiroga

Integrantes:

| | |
|---|---------------------|
| <i>Chambilla Mardínez, Renato Eduardo</i> | <i>(2020066918)</i> |
| <i>Medina López, Marcelo José</i> | <i>(2020066917)</i> |
| <i>Valle Bustamante, Gustavo Alonso</i> | <i>(2020066916)</i> |

Tacna – Perú

2024

Proyecto *RenatitoBot*

Documento de Arquitectura de Software

Versión *1.0*

| CONTROL DE VERSIONES | | | | | |
|----------------------|---|--------------------------------------|--------------|------------|-----------|
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | Renato Chambilla Mardinez, Gustavo Valle Bustamante, Marcelo Medina López | Ing. Patrick José Cuadros Quiroga | | 07/07/2024 | Versión 1 |

INDICE GENERAL

| | |
|---|-----------|
| 1. INTRODUCCIÓN | 4 |
| 1.1. Propósito | 4 |
| 1.2. Alcance | 4 |
| 1.3. Definición, siglas y abreviaturas | 4 |
| 2. OBJETIVOS Y RESTRICCIONES ARQUITECTONICAS | 4 |
| 2.1.1. Requerimientos Funcionales..... | 4 |
| 2.1.2. Requerimientos No Funcionales – Atributos de Calidad | 5 |
| 3. REPRESENTACIÓN DE LA ARQUITECTURA DEL SISTEMA | 6 |
| 3.1. Vista de Caso de uso | 6 |
| 3.1.1. Diagramas de Casos de uso | 6 |
| 3.2. Vista Lógica..... | 7 |
| 3.2.1. Diagrama de Subsistemas (paquetes)..... | 7 |
| 3.2.2. Diagrama de Secuencia (vista de diseño)..... | 8 |
| 3.2.3. Diagrama de Objetos..... | 10 |
| 3.2.4. Diagrama de Clases..... | 11 |
| Figura 11: Diagrama de Clases. Fuente de origen: Propia..... | 11 |
| 3.3. Vista de Implementación (vista de desarrollo) | 12 |
| 3.3.1. Diagrama de arquitectura software (paquetes) | 12 |
| 3.3.2. Diagrama de arquitectura del sistema (Diagrama de componentes)..... | 13 |
| 3.4. Vista de procesos | 13 |
| 3.4.1. Diagrama de Procesos del sistema (diagrama de actividad) | 13 |
| 3.5. Vista de Despliegue (vista física) | 15 |
| 3.5.1. Diagrama de despliegue | 15 |
| 4. ATRIBUTOS DE CALIDAD DEL SOFTWARE..... | 15 |
| Escenario de Funcionalidad | 15 |
| Escenario de Usabilidad..... | 16 |
| Escenario de confiabilidad | 16 |
| Escenario de rendimiento | 16 |
| Escenario de mantenibilidad..... | 16 |
| Otros Escenarios..... | 16 |

1. INTRODUCCIÓN

1.1. Propósito

El propósito de este proyecto es desarrollar un juego 3D inmersivo donde los jugadores encarnen a un robot explorador en una isla misteriosa, enfrentándose a desafíos que no solo implican exploración, sino también la resolución de preguntas y enigmas científicos. A través de esta experiencia interactiva, el juego busca no solo entretener, sino también educar, promoviendo el aprendizaje y la curiosidad científica. El objetivo es ofrecer una plataforma que combine el disfrute visual con el estímulo intelectual, incentivando a los jugadores a descubrir nuevos lugares y desbloquear habilidades mientras exploran este intrigante entorno virtual.

1.2. Alcance

Queremos que nuestro juego no solo sea aceptado por la comunidad, sino que también sea aclamado por su innovación y creatividad. Nos esforzamos por ofrecer una experiencia de juego única y emocionante que sorprenda a los jugadores con mecánicas innovadoras, gráficos impresionantes y una historia envolvente. Además, estamos comprometidos a mantener una estrecha relación con nuestra comunidad, escuchando sus comentarios y sugerencias para mejorar continuamente la experiencia del juego.

1.3. Definición, siglas y abreviaturas

- Videojuego: Es un juego electrónico que implica la interacción con una interfaz de usuario o un dispositivo de entrada para generar retroalimentación visual desde un dispositivo.

2. OBJETIVOS Y RESTRICCIONES ARQUITECTONICAS

2.1. Priorización de requerimientos

2.1.1. Requerimientos Funcionales

Tabla 1: Cuadro de Requerimientos Funcionales Final. Fuente de origen: Propia.

| Número | Requerimiento Funcional | Casos de uso | Prioridad |
|---------|---|----------------|-----------|
| RF - 01 | El videojuego deberá tener una presentación inicial al iniciarse. | Mostrar Splash | 1 |

| | | | |
|---------|--|-------------------------------|---|
| RF - 02 | El videojuego deberá tener una sección de opciones incluida en el splash. | Mostrar Menú de Opciones | 1 |
| RF - 03 | Dentro del menú de opciones se debe permitiri manejar configuraciones del videojuego, como calidad, sonido o cualquier aspecto. | Permitir Configuraciones | 1 |
| RF – 04 | El videojuego deberá tener por lo menos dos niveles. | Presentar dos niveles | 1 |
| RF – 05 | El videojuego permitirá al usuario mover al personaje en todas las direcciones. | Controlar Personaje | 1 |
| RF – 06 | El videojuego deberá mostrar preguntas al jugador para progresar en los niveles. | Mostrar Preguntas | 2 |
| RF – 07 | El videojuego finalizará los niveles cuando se hayan respondido todas las preguntas. | Finalizar Niveles | 2 |
| RF – 08 | El videojuego mostrará un mensaje de bienvenida al empezar el primer nivel. | Mostrar Mensaje de Bienvenida | 3 |
| RF – 09 | El videojuego mostrará un mensaje de bienvenida al terminar todos los niveles. | Mostrar Mensaje de Victoria. | 3 |
| RF – 10 | El videojuego mostrará objetos brillantes que servirán para mostrar las preguntas. | Mostrar Objetos | 3 |
| RF – 11 | El videojuego mostrará en un texto ubicado en la parte superior de la pantalla la cantidad de vidas del jugador, la cual disminuirá si se responde mal las preguntas. | Mostrar Cantidad de Vidas | 3 |
| RF-12 | El videojuego mostrará en un texto ubicado en la parte superior de la pantalla la cantidad de vidas del jugador, la cual disminuirá si se responden correctamente las preguntas. | Mostrar Cantidad de Preguntas | 3 |

Descripción: La tabla 1 muestra el Cuadro de Requerimientos Funcionales Final para el sistema propuesto. Cada requerimiento describe una función específica que se espera que el videojuego tenga, junto con su correspondiente caso de uso y prioridad asignada. Estos requisitos abarcan aspectos que debe tener el videojuego de comienzo a fin.

2.1.2. Requerimientos No Funcionales – Atributos de Calidad

Tabla 2: Cuadro de Requerimientos No Funcionales. Fuente de origen: Propia.

| Número | Requerimiento No Funcional | Descripción | Prioridad |
|----------|----------------------------|--|-----------|
| RNF - 01 | Seguridad | La información o datos de los usuarios debe estar protegida. | 3 |
| RNF - 02 | Disponibilidad | El videojuego debe estar accesible en todo momento y el personal de soporte debe ayudar en caso de que no. | 3 |
| RNF - 03 | Usabilidad | El videojuego debe ser de sencillo entendimiento y manipulación para los usuarios. | 2 |
| RNF - 04 | Fiabilidad | Los usuarios deben sentirse cómodos con el videojuego | 2 |
| RNF - 05 | Mantenibilidad | El videojuego no debería ser tan complicada de actualizar. | 2 |

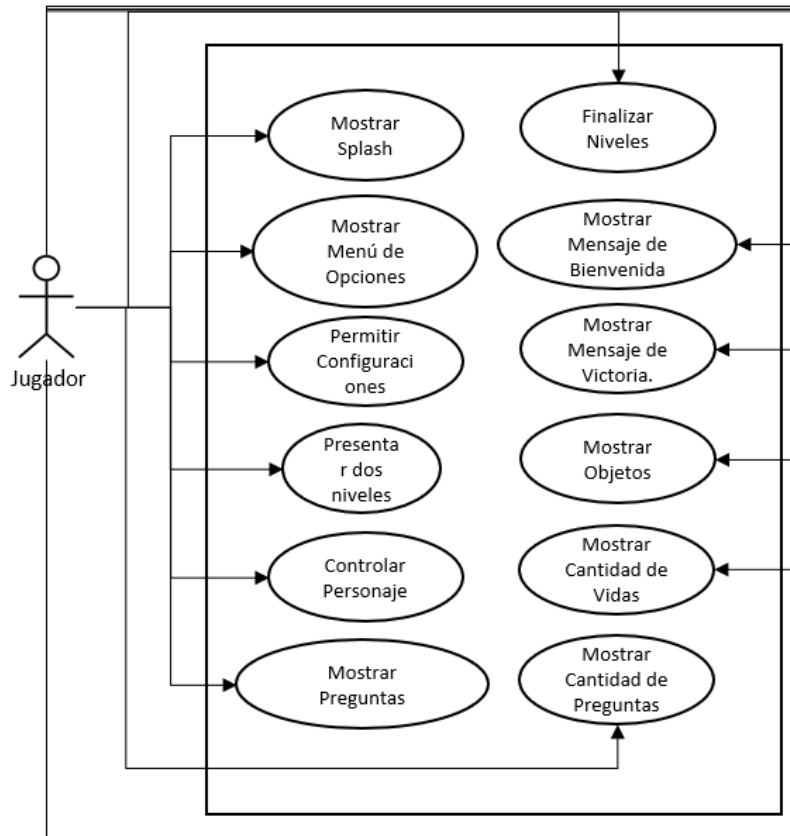
Descripción: La tabla 2 presenta un Cuadro de Requerimientos No Funcionales para el videojuego propuesto. Estos requisitos describen aspectos importantes relacionados con la seguridad, disponibilidad, usabilidad, fiabilidad y mantenibilidad del sistema. Cada requisito está numerado y se proporciona una breve descripción de lo que implica. La prioridad asignada a cada requisito indica su importancia relativa en el contexto del desarrollo y mantenimiento del videojuego. Estos requisitos no se centran en las funciones específicas del sistema, sino en cómo debe comportarse y operar el sistema en términos de características generales, rendimiento y mantenimiento.

3. REPRESENTACIÓN DE LA ARQUITECTURA DEL SISTEMA

3.1. Vista de Caso de uso

3.1.1. Diagramas de Casos de uso

Figura 1: Diagrama de Casos de Uso. Fuente de origen: Propia.



Descripción: La Figura 1 representa un diagrama de los casos de uso planteados y su interacción con los actores examinados dentro del proceso, siendo el caso del proyecto únicamente de los jugadores como únicos actores.

3.2. Vista Lógica

3.2.1. Diagrama de Subsistemas (paquetes)

Figura 2: Diagrama de Paquetes. Fuente de origen: Propia.

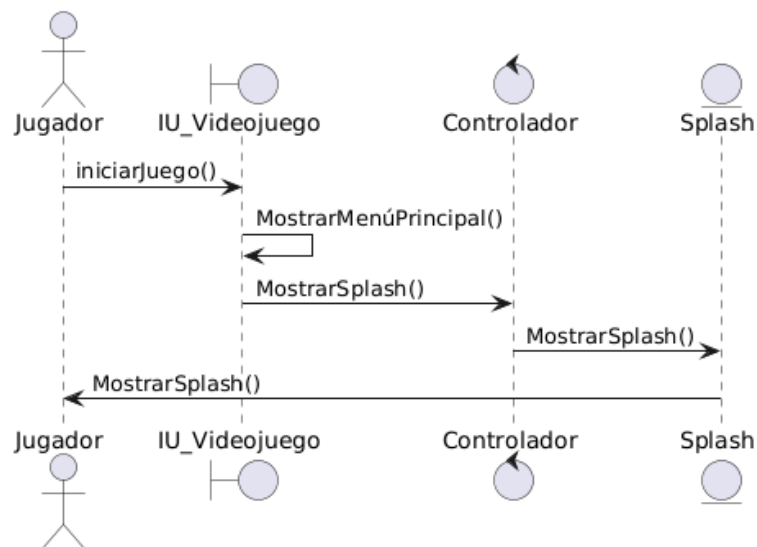


Descripción: La Figura 2 representa un diagrama de paquetes que muestra la estructura de los distintos paquetes del videojuego. Cada paquete representa un conjunto de funcionalidades relacionadas que se agrupan de manera lógica y coherente para facilitar la comprensión y el desarrollo del videojuego.

3.2.2. Diagrama de Secuencia (vista de diseño)

- Mostrar Splash:

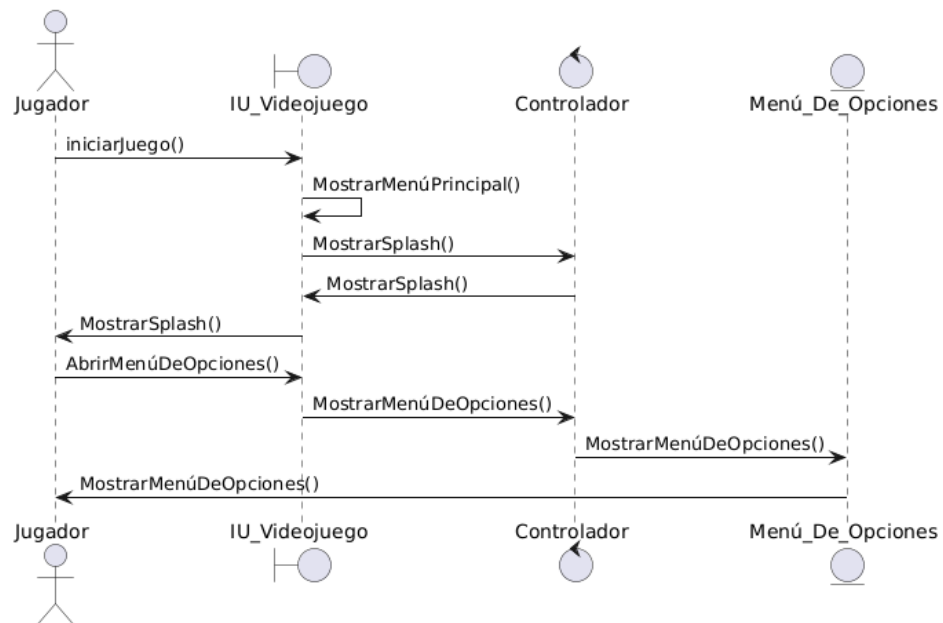
Figura 3: Diagrama de secuencia del caso de uso: Mostrar Splash. Fuente de origen: Propia



Descripción: La figura 3 detalla los pasos dentro del proceso del caso de uso “Mostrar Splash”, donde se destacan las acciones del actor y los objetos correspondientes.

- Mostrar Menú de Opciones

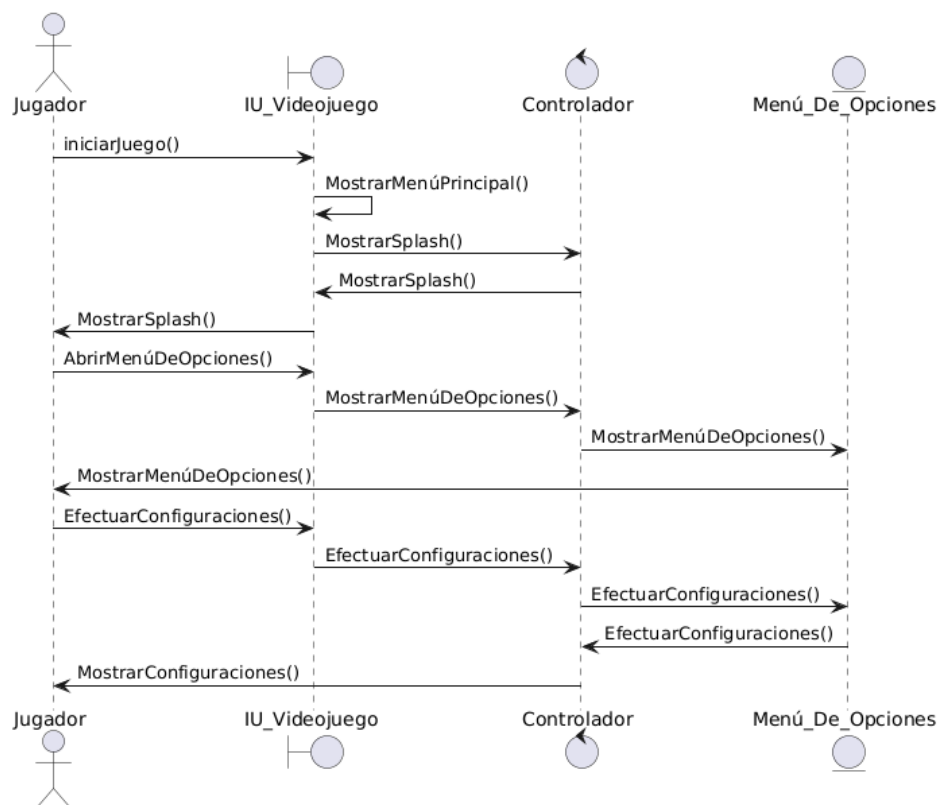
Figura 4: Diagrama de secuencia del caso de uso: Mostrar Menú de Opciones. Fuente de origen: Propia



Descripción: La Figura 4 detalla los pasos dentro del proceso del caso de uso “Mostrar Menú de Opciones”, donde se destacan las acciones del actor y los objetos correspondientes.

- Permitir Configuraciones

Figura 5: Diagrama de secuencia del caso de uso: Permitir Configuraciones. Fuente de origen: Propia

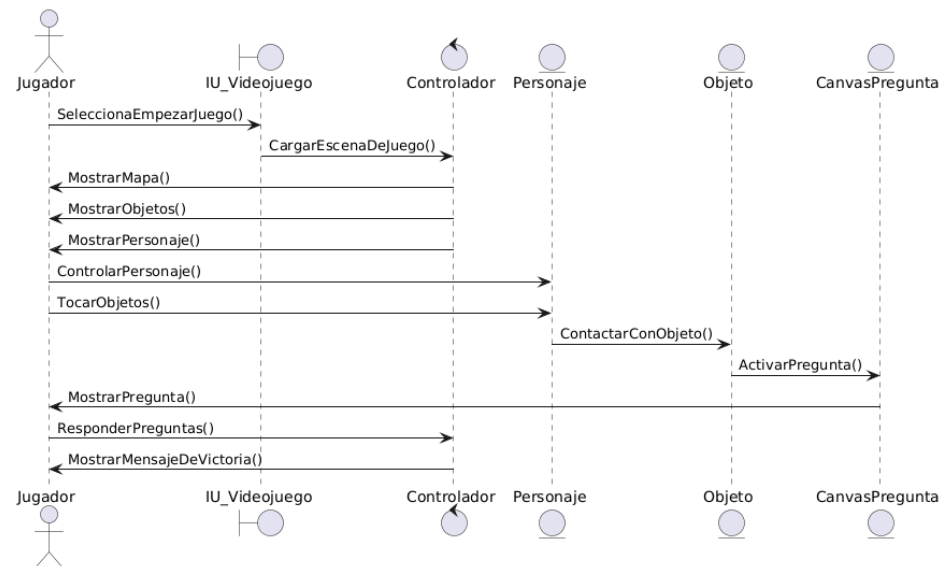


Descripción: La Figura 5 detalla los pasos dentro del proceso del caso de uso “Permitir Configuraciones”, donde se destacan las acciones del actor y los objetos correspondientes.

- Presentar Dos Niveles:

Figura 6: Diagrama de secuencia del caso de uso: Presentar Dos Niveles.

Fuente de origen: Propia

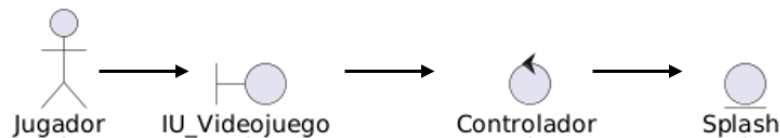


Descripción: La Figura 6 detalla los pasos dentro del proceso del caso de uso “Presentar Dos Niveles”, donde se destacan las acciones del actor y los objetos correspondientes.

3.2.3. Diagrama de Objetos

- Mostrar Splash:

Figura 7: Análisis de Objetos del Caso de Uso: Mostrar Splash. Fuente de origen: Propia

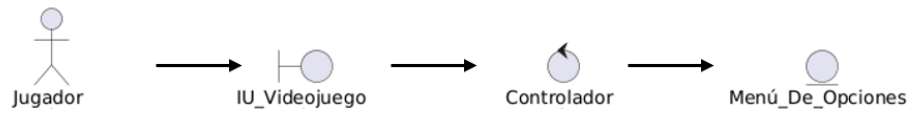


Descripción: La Figura 7 describe gráficamente que actores y objetos interactuarán en el presente caso de uso.

- Mostrar Menú de Opciones:

Figura 8: Análisis de Objetos del Caso de Uso: Mostrar Menú de Opciones.

Fuente de origen: Propia

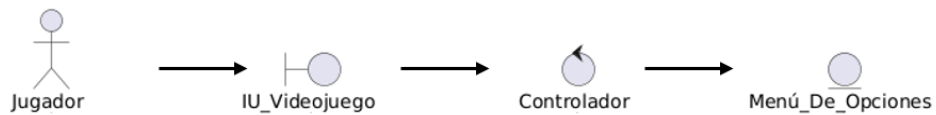


Descripción: La Figura 8 describe gráficamente que actores y objetos interactuarán en el presente caso de uso.

- Permitir Configuraciones:

Figura 9: Análisis de Objetos del Caso de Uso: Permitir Configuraciones.

Fuente de origen: Propia

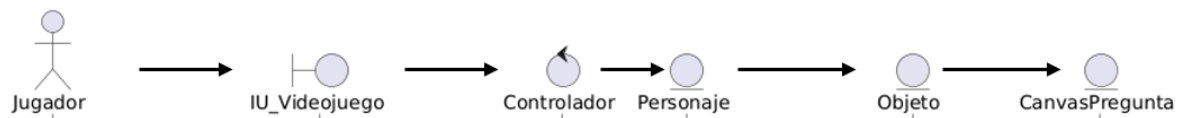


Descripción: La Figura 9 describe gráficamente que actores y objetos interactuarán en el presente caso de uso.

- Presentar Dos Niveles:

Figura 10: Análisis de Objetos del Caso de Uso: Presentar Dos Niveles.

Fuente de origen: Propia



Descripción: La Figura 10 describe gráficamente que actores y objetos interactuarán en el presente caso de uso.

3.2.4. Diagrama de Clases

Figura 11: Diagrama de Clases. Fuente de origen: Propia

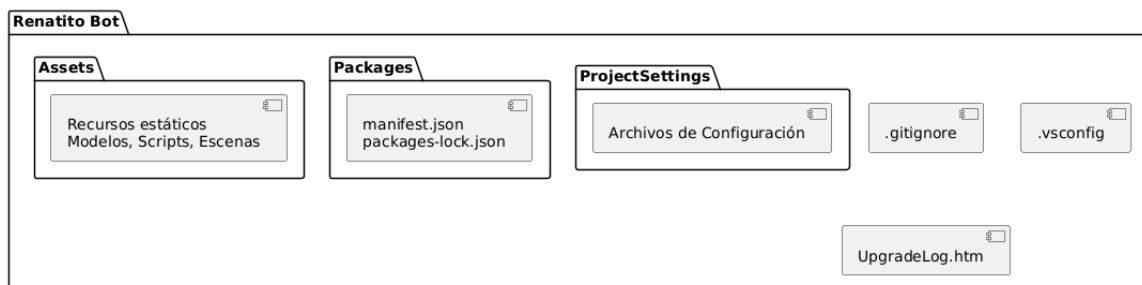


Descripción: La Figura 11 muestra la conexión entre las diferentes clases del videojuego, dando una leve muestra de los métodos atribuidos a cada una.

3.3. Vista de Implementación (vista de desarrollo)

3.3.1. Diagrama de arquitectura software (paquetes)

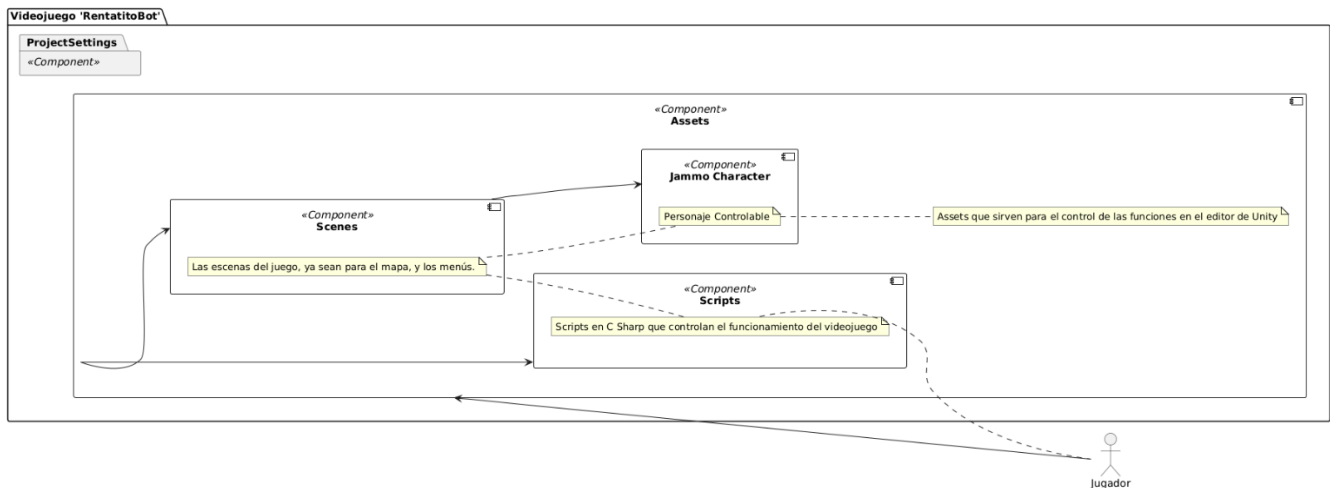
Figura 12: Diagrama de Paquetes. Fuente de origen: Propia.



Descripción: La Figura 12 representa un diagrama de paquetes que muestra la estructura de los distintos paquetes del videojuego. Cada paquete representa un conjunto de funcionalidades relacionadas que se agrupan de manera lógica y coherente para facilitar la comprensión y el desarrollo del videojuego.

3.3.2. Diagrama de arquitectura del sistema (Diagrama de componentes)

Figura 13: Diagrama de Componentes. Fuente de origen: Propia.

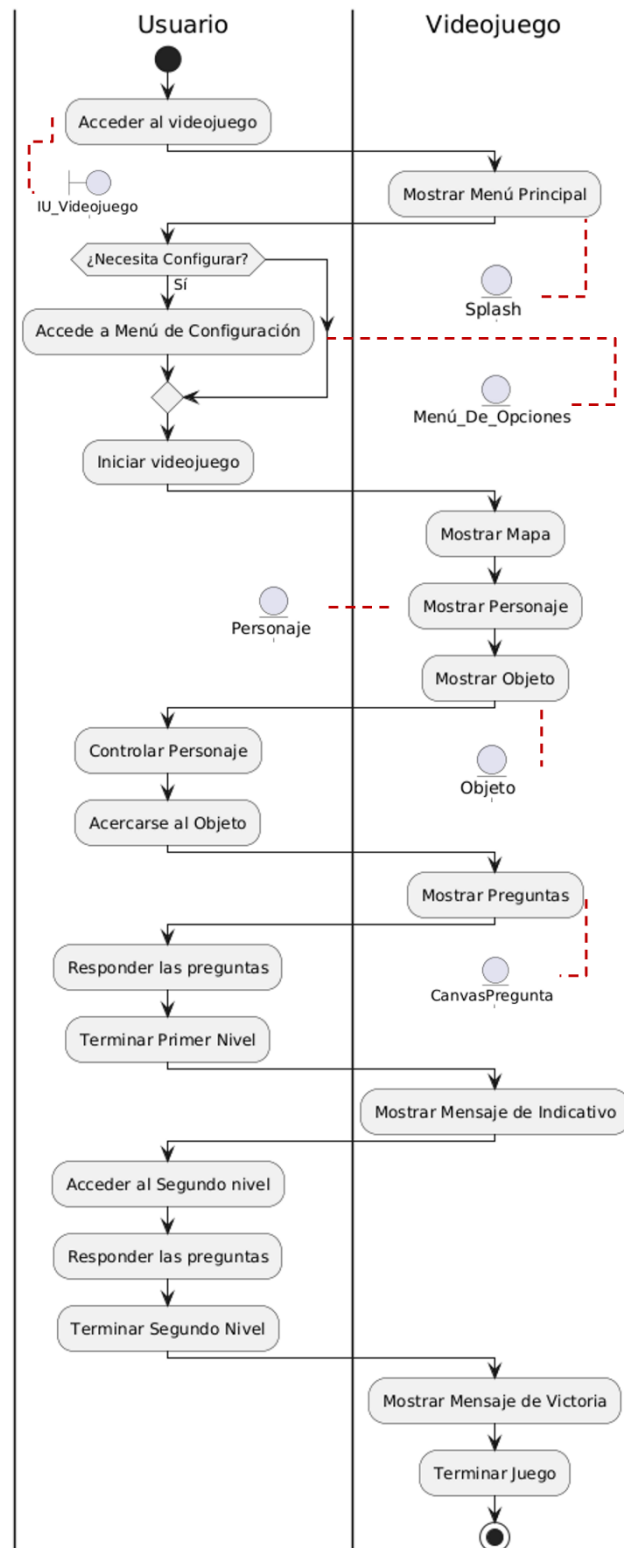


Descripción: La Figura 13 muestra la estructura principal del juego desarrollado en Unity. El jugador interactúa con el paquete "Assets", que contiene los componentes "Scripts" (que controlan el funcionamiento del videojuego mediante scripts en C#), "Scenes" (que incluyen las escenas del juego como mapas y menús), y "Jammo Character" (el personaje controlable). Adicionalmente, el paquete "ProjectSettings" contiene "Assets" específicos para controlar las funciones en el editor de Unity.

3.4. Vista de procesos

3.4.1. Diagrama de Procesos del sistema (diagrama de actividad)

Figura 14: Diagrama de Actividades con objetos: Proceso de Autenticación.
Fuente de origen: Propia

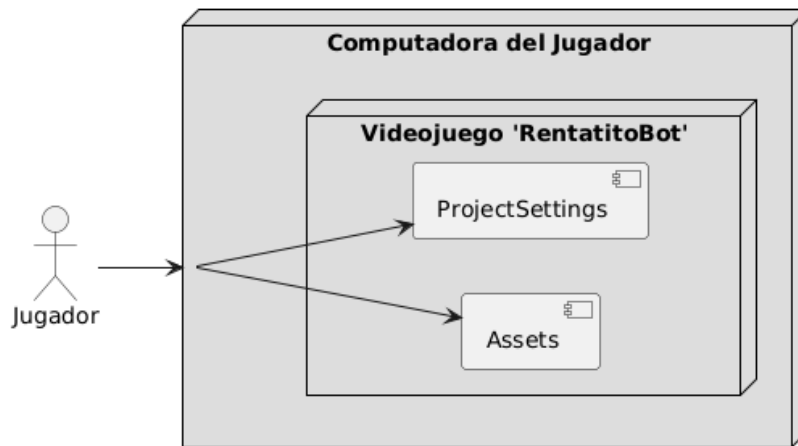


Descripción: La Figura 14 representa un Diagrama de Actividades del proceso principal del videojuego en sí, con sus respectivos objetos entidad ML. En la presente figura se ilustran procesos dese que el jugador ejecuta el videojuego hasta que lo termina.

3.5. Vista de Despliegue (vista física)

3.5.1. Diagrama de despliegue

Figura 15: Diagrama de Actividades con objetos: Proceso de Autenticación.
Fuente de origen: Propia



Descripción: La Figura 15 ilustra cómo se despliega el juego en la computadora del jugador. El jugador interactúa con el videojuego que se ejecuta en su computadora. Dentro de esta computadora, el videojuego "RentatitoBot" se compone de los componentes "Assets" y "ProjectSettings", necesarios para el funcionamiento y la configuración del juego en el entorno de Unity. Este diagrama destaca la relación entre el jugador y la infraestructura del juego en su máquina local.

4. ATRIBUTOS DE CALIDAD DEL SOFTWARE

Escenario de Funcionalidad

El juego debe ofrecer un conjunto diverso y robusto de características y capacidades. Esto incluye la implementación fluida de mecánicas de exploración, resolución de enigmas científicos, y narrativa interactiva en un entorno 3D inmersivo. La funcionalidad también debe garantizar la seguridad del sistema contra posibles vulnerabilidades y asegurar que todas las interacciones del jugador con el entorno virtual sean coherentes y estables.

Escenario de Usabilidad

Es fundamental que el juego sea fácil de aprender y utilizar para los jugadores. Esto implica una interfaz intuitiva que permita a los jugadores navegar por el entorno, interactuar con objetos y resolver enigmas de manera eficiente. La usabilidad debe minimizar el impacto de errores, adaptarse a las preferencias individuales de los usuarios y garantizar una experiencia de juego satisfactoria y confiable que fomente la confianza en la plataforma educativa y de entretenimiento.

Escenario de confiabilidad

El juego debe asegurar la confidencialidad, integridad, irrefutabilidad y disponibilidad de la información y datos manipulados dentro del sistema. Esto implica implementar medidas de seguridad robustas para proteger los datos del usuario y el funcionamiento del juego contra amenazas físicas, lógicas y humanas. Además, se deben establecer mecanismos de prevención, precaución y reacción ante posibles vulnerabilidades y ataques, garantizando un entorno seguro y confiable para los jugadores.

Escenario de rendimiento

El rendimiento del juego se evaluará en términos de velocidad de procesamiento, tiempo de respuesta, uso eficiente de recursos y capacidad para manejar una carga de trabajo variada. El juego debe ser capaz de ofrecer una experiencia fluida y sin interrupciones, incluso durante las secuencias intensivas de gráficos 3D y procesamiento de datos asociados con la exploración y la resolución de enigmas. Esto incluye mantener un rendimiento óptimo en diversas plataformas de juego, como PC, consolas y dispositivos móviles.

Escenario de mantenibilidad

La mantenibilidad del juego se centra en su capacidad para ser ampliable, adaptable y mantenible a largo plazo. Esto implica utilizar prácticas de desarrollo de software que faciliten la incorporación de nuevas características y contenidos educativos, así como la corrección de errores y la optimización del rendimiento. La estructura del código y la documentación deben apoyar la escalabilidad del juego y su capacidad para evolucionar con actualizaciones continuas y expansiones de contenido.

Otros Escenarios

Performance: El escenario adicional de Performance se refiere a la capacidad del juego para responder eficientemente a eventos y procesar una cantidad adecuada de eventos en un intervalo de tiempo determinado. Esto incluye optimizar los tiempos de carga, la fluidez de las animaciones y la capacidad de renderizado para garantizar una experiencia de juego inmersiva y sin interrupciones, mejorando así la satisfacción del jugador y la percepción general de calidad del producto.