**Universidad Autónoma del Estado de México**

**Unidad Académica Profesional Tianguistenco**

**Ingeniería en software**

**Unidad de aprendizaje:**

**Graficación**

**Docente:**

**Elizabet Pulido**

**Título:**

**Aerix**

**Integrantes:**

Carlos Jonathan López Palma

David Velázquez Ramírez

Alan Alexis Velázquez Romero

Arely Lara Silva

Fecha de entrega: 14/04/2020

Contenido

[Introducción 3](#_Toc40272946)

[Problemática 4](#_Toc40272947)

[**Planteamiento del problema** 4](#_Toc40272948)

[**Contexto** 4](#_Toc40272949)

[**Problemática** 4](#_Toc40272950)

[**Propuesta de solución** 4](#_Toc40272951)

[**Delimitación** 4](#_Toc40272952)

[**Justificación** 5](#_Toc40272953)

[**Objetivo General** 5](#_Toc40272954)

[**Objetivos Específicos** 5](#_Toc40272955)

[Marco teórico 5](#_Toc40272956)

[**Estadísticas de los videojuegos en México** 7](#_Toc40272957)

[**Análisis FODA** 7](#_Toc40272958)

[Propuesta de solución 8](#_Toc40272959)

[**Requerimientos funcionales y no Funcionales** 8](#_Toc40272960)

[**Ciclo de vida del software** 8](#_Toc40272961)

[**Historia del juego** 11](#_Toc40272962)

[Descripción de los personajes 11](#_Toc40272963)

[Personajes 3D 12](#_Toc40272964)

[**Análisis de requerimientos** 14](#_Toc40272965)

[**Casos de uso** 28](#_Toc40272966)

[Bibliografía 36](#_Toc40272967)

[Anexos 36](#_Toc40272968)

**Índice de Figuras**

[1. Modelo en Espiral 9](#_Toc37788410)

[2. Mago Soporte 11](#_Toc37788411)

[3. Aria 11](#_Toc37788412)

[4. Flogats 11](#_Toc37788413)

[5. Terluks 12](#_Toc37788414)

# Introducción

Para mantenerse seguro, en la comodidad de la casa y como forma de entretenimiento, los videojuegos son una gran alternativa para mantenerte distraído, por lo que este proyecto se enfoca en la creación de un videojuego de poca dificultad, que cualquier persona de cualquier edad pueda disfrutar y entender.

De igual forma, utilizar este medio para educar e introducir a los videojuegos a las personas que no poseen conocimiento de este mundo y más aún, transmitir aquellos conceptos poco conocidos utilizados en esta área. De forma personal el conocer como es el trabajo necesario para desarrollar un videojuego y aprender a valorar el trabajo de aquellas personas que se dedican a este medio.

Abarcaremos los principios básicos que se llevaran a cabo para la elaboración del juego, tratando el uso de una pequeña historia pulida a mano, el diseño de personajes y escenarios, la banda sonora que será utilizada para ambientar dicho juego, así como las mecánicas que se implementaran para interactuar con el juego en sí.

# Problemática

## **Planteamiento del problema**

### **Contexto**

Actualmente vivimos en una época en el que el mundo es cada vez más peligroso, en el que salir a divertirte presenta un riego para la salud e integridad de las personas. De la misma forma, esta época se caracteriza por contar con una gran cantidad de herramientas de distracción, smartphones, tablets, computadoras, internet, etc., y aunque cumplen su función, la necesidad de salir a convivir con amigos, ir al cine, a comer o cualquier actividad al exterior es mayor para los jóvenes, que requieren mantenerse distraídos y en constante movimiento.

### **Problemática**

Para enfrentar esta situación en la que debemos mantenernos sin salir de nuestros hogares, el aburrimiento se ha convertido en un gran problema para la población que no puede relajarse estando encerrado y que la ansiedad le provoca la necesidad de salir de casa.

### **Propuesta de solución**

Como medida de solución, para que las personas puedan quedarse en casa sin ser presar del aburrimiento y la amplia cantidad de personas que pueden o saben utilizar una computadora, surge la idea de crear un videojuego para esta plataforma, que ayude a las personas a evitar sentirse agobiados o aburridos.

## **Delimitación**

Este videojuego está enfocado para personas de cualquier edad, que sea fácil de entender para cualquiera de ellos y, aunque no tenga conocimiento de conceptos básicos dentro de los videojuegos, pueda desenvolverse y jugar sin problema.

De igual forma que cuente con una historia simple y una narrativa nada compleja, para que el jugador pueda seguir el curso de esta historia sin perderse.

## **Justificación**

Es necesario hacer un videojuego fácil de jugar y entender, pero con una trama interesante y lo suficientemente compleja, para que tanto, personas con conocimiento en videojuegos e historias fantásticas, como para los principiantes, se diviertan.

## **Objetivo General**

Crear un videojuego de dificultad media, entretenido, divertido y con una historia que atrape a todas las personas.

## **Objetivos Específicos**

* Amigable con el jugador.
* Pocos requerimientos gráficos.
* No es un juego bélico
* Temática simple

# Marco teórico

**Gráficos.** Son representaciones visuales sobre alguna superficie, como lienzo (canvas), pantallas o papel. Los gráficos son usados para informar, ilustrar o entretener. Ejemplos de gráficos son: pinturas, fotografías, dibujos, imágenes digitales, diagramas, diseños geométricos, mapas. Los gráficos pueden ser en blanco y negro, escala de grises o color, y pueden contener texto.

**Píxel**. Es la unidad más pequeña de una imagen digital. En la computadora, un pixel se representa por su código de color, el cual es, o bien un valor en la escala de grises o los tres componentes de un color. Tendemos a pensar que un pixel es un pequeño punto, circular o cuadrado, sin embargo, un pixel es matemáticamente un punto sin dimensiones.

**Dimensiones.** Las dimensiones definen el espacio en que un objeto puede existir.

**Computación gráfica 2D.** Corresponde al conjunto de técnicas que tienen como objeto la generación de una imagen digital a partir de modelos geométricos bidimensionales.

**Computación gráfica 3D**. Trata acerca de la síntesis de una imagen bidimensional a partir de un modelo geométrico tridimensional. De acuerdo con la complejidad de los cálculos empleados en la generación de las imágenes, las técnicas se clasifican en prerendering y real-time rendering.

**Gráfica interactiva**. Es la técnica de utilizar comunicación bidireccional entre la computadora y el usuario en la que la computadora se comunica con imágenes gráficas.

**Gráfica pasiva**. La gráfica pasiva es simplemente una representación de datos en una manera visual, sin alguna interacción entre el programa y el usuario.

**Videojuego**. Es un juego electrónico en el que una o más personas interactúan, por medio de un controlador, con un dispositivo que muestra imágenes de video. Este dispositivo electrónico, conocido genéricamente como «plataforma», puede ser una computadora, una máquina arcade, una videoconsola o un dispositivo portátil.

**Formato**. Es un estándar que define la forma en que la información se organiza y se codifica en un archivo informático. Dado que una unidad de disco o de cualquier memoria solo puede almacenar bits, el ordenador o computadora debe tener algún método para convertir la información en datos binarios (ceros y unos).

**JPEG.** Es un mecanismo estandarizado de compresión de imágenes. JPEG son las siglas para Joint Photographic Experts Group, el nombre original del comité que escribió el estándar. JPEG está diseñado para comprimir ya sea imágenes a color o en escala de gris de escenas naturales y realistas.

**Transformaciones.** Una vez que una imagen ha sido generada, el usuario con frecuencia la transforma para ver diferentes partes de la imagen, o para modificar su aspecto en formas regulares. Las transformaciones son una necesidad cuando una imagen tridimensional es vista en una pantalla bidimensional.

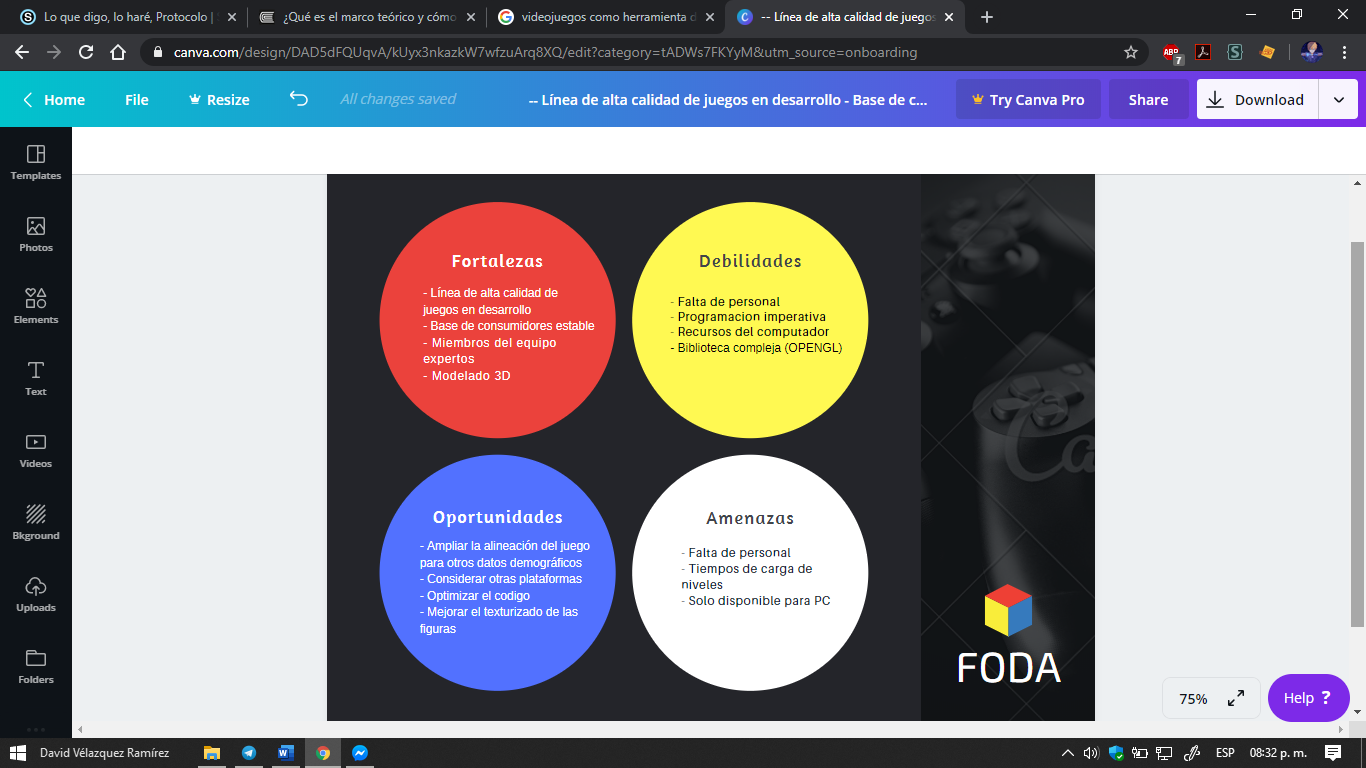
**Proyecciones.** Una imagen tridimensional tiene que ser proyectada para imprimirla o verla en una pantalla bidimensional. La proyección más común es la perspectiva.

**Rendering** (Representar, Interpretar)). El proceso de generar una imagen digital o gráfico rasterizado, a partir de un modelo matemático, por medio de un algoritmo implementado en programas de cómputo. El modelo matemático describe los objetos que constituyen la imagen en términos de puntos y curvas.

## **Estadísticas de los videojuegos en México**



## **Análisis FODA**



# Matrices 3D

Se hace uso de matrices para la generación de objetos 3D utilizando las herramientas que nos ofrece OpenGL como lo son GLU y GLUT. Dichas matrices son generadas en los Render de cada objeto que se utilizo en el juego, los personajes hacen uso de funciones como GLPushMatrix, GLPopMatrix, etc.

La inclusión de las transformaciones tridimensionales es de gran ayuda pues permiten trasladar, escalar y rotar los objetos del juego.

# OpenGL y Java

OpenGL (Open Graphics Library) es una especificación estándar que define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D. La interfaz consiste en más de 250 funciones diferentes que pueden usarse para dibujar escenas tridimensionales complejas a partir de primitivas geométricas simples, tales como puntos, líneas y triángulos.

Java es un lenguaje de programación y una plataforma informática que fue comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán, probablemente, a menos que tengan Java instalado y cada día se crean más. Java es rápido, seguro y fiable.

# Texturización

Una textura, desde el punto de vista de su almacenamiento en memoria es un array de datos. Cada uno de los valores de este array lo llamamos 'textel'. Este array de datos representa una imagen, que utilizaremos para mapearla sobre un polígono.

Podemos rellenar dicho array bien cargando una imagen desde un fichero, o bien dándole nosotros los valores de color a cada texel.

La información de cada uno de los componentes puede ser:

* Las componentes RGB del color.
* Índices de color.
* Niveles de luminancia (grises).

Además, podemos tener texturas unidimensionales (un solo píxel de alto o de ancho) bidimensionales (imagen de tamaño mxn) o tridimensionales (con volumen), aunque lo habitual será utilizar las bidimensionales.

# Colisiones

Decimos que dos objetos colisionan cuando uno de ellos se sobrepone a otro en este momento debemos disparar una "señal" y tratar dicha colisión en consecuencia, impidiendo el movimiento si es un sólido, restando vida si es un enemigo, etc. Todo dependerá del tipo de juego en este artículo nos vamos a centrar en detectar dichas colisiones.

# Cámara

Es la región cerrada del espacio que delimita los objetos que aparecen representados en la pantalla. De acuerdo con la geometría de la cámara virtual que sirve para visualizar las imágenes de objetos cuyas coordenadas se conocen, delimita la zona visible que aparece en las imágenes generadas.

# Sonido

El sonido o música es, tal vez, el elemento ambiental más sencillo de implementar en la partida, en cuanto a que basta con cualquiera de los múltiples medios de que disponemos para reproducirla. Y por eso es algo que no suele faltar en las partidas. Pero puede convertirse en un arma de doble filo, ya que mal utilizada puede arruinarnos el juego.

La idea clave es que la música tiene que ser una banda sonora. Tiene que pasar casi desapercibida. Y tiene que ir en función de cada escena. La música debe servir para acompañar, para ambientar. No estamos en un concierto, estamos en una partida de rol. Si no va a cumplir esta función, es mejor quitarla.

# Propuesta de solución

## **Requerimientos funcionales y no Funcionales**

RF

* Movimientos del personaje
* Lectura de teclado
* Seleccionar nivel
* Seleccionar personaje
* Configurar
* Créditos
* Cargar nivel
* Pausar nivel
* Movimiento de camara

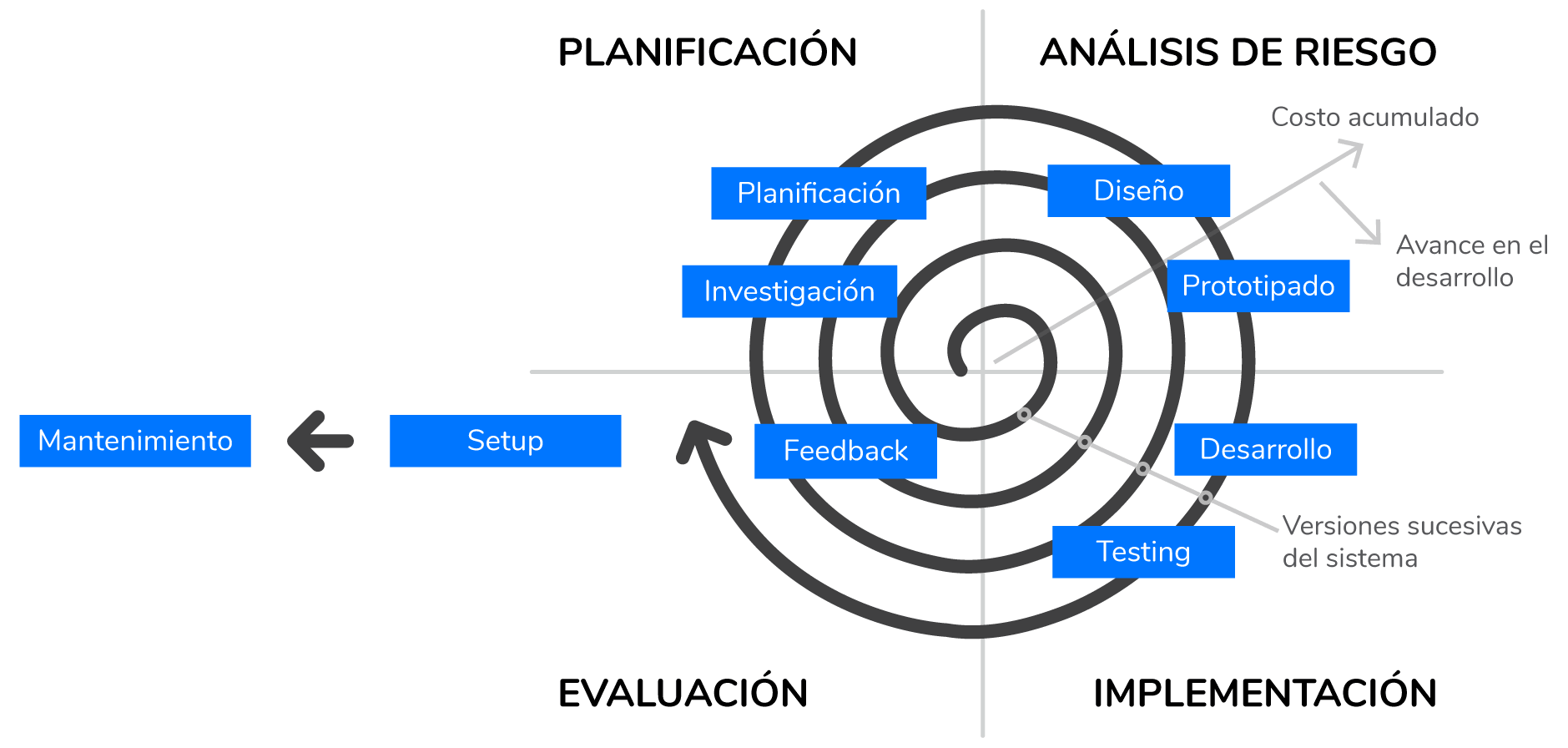
RNF

* Buen sonido
* Colores agradables
* Historia entretenida
* Buena resolución
* Textos visibles
* Sin lag

## **Ciclo de vida del software**

**Análisis de los requisitos y su viabilidad**: recopila, examina y formula los requisitos del cliente y examina cualquier restricción que se pueda aplicar.  
  
**Diseño general**: requisitos generales de la arquitectura de la aplicación.  
  
**Diseño en detalle**: definición precisa de cada subconjunto de la aplicación.  
  
**Programación** (programación e implementación): implementación de un lenguaje de programación para crear las funciones definidas durante la etapa de diseño.  
  
**Prueba de unidad**: prueba individual de cada subconjunto de la aplicación para garantizar que se implementaron de acuerdo con las especificaciones.  
  
**Integración**: garantiza que los diferentes módulos se integren con la aplicación. Este es el propósito de la prueba de integración que está cuidadosamente documentada.  
  
**Prueba beta** (o validación): garantiza que el *software* cumple con las especificaciones originales.  
  
**Documentación**: sirve para documentar información necesaria para los usuarios del *software* y para desarrollos futuros.  
  
**Implementación**  
  
**Mantenimiento**: comprende todos los procedimientos correctivos (mantenimiento correctivo) y las actualizaciones secundarias del *software* (mantenimiento continuo).

**Modelo en espiral**



1. Modelo en Espiral

1. **Planificación**. Se determinan los objetivos y el alcance del ciclo que comienza, tras un necesario ejercicio de investigación. Con cada iteración, se irá incrementando el tamaño de software entregado y la funcionalidad cubierta.
2. **Análisis de Riesgo.** Se evalúa todo aquello que pueda afectar al proyecto según el estado en que se encuentre y su grado de avance. Para ello, se diseñarán los prototipos que deberán ser validados en el ciclo.
3. **Implementación**. Se desarrolla y valida el software según el alcance acordado, el cual está íntimamente relacionado y condicionado con el análisis de riesgos anterior.
4. **Evaluación**. Antes de proceder a realizar otra vuelta en la espiral, se debe prestar atención a lo que sucedió en la vuelta anterior. Se debe analizar en detalle si los riesgos detectados anteriormente ya tuvieron solución. Básicamente, esta fase servirá para determinar el avance del proyecto y dar pistas de hacia dónde debe enfocarse la próxima iteración.

## **Historia del juego**

**Introducción al viaje.**

La historia se remonta en una pequeña ciudad del medioevo llamada *AERIX*, en esta ciudad la magia ha sido posible gracias a la obtención de un mineral conocido como *LUKTITA*, este extraño mineral solo pudo ser extraído en una mina a lo lejos de la ciudad, el mineral tiene la capacidad de desprender una esencia que permite a los usuarios obtener mejoras físicas siempre y cuando este mineral sea tratado y contenido en lo que se le conoce como *KEYLUKS*, cierto día la población de *AERIX* comenzó a decaer debido a las inexplicables muertes de los ciudadanos que poco a poco iban propagándose en cadena, dejando a la ciudad sin la oportunidad de estudiar la enfermedad y sin dicho mineral, es aquí donde un grupo de expertos conocidos como *FRAGMA* emprenden una búsqueda para la recolección de *LUKTITA* para generar una posible cura, pero el camino está lleno de peligros y trampas en la mina.

**Lo KEYLUKS**

Son objetos creados a base de metales preciosos y luktita en estado puro, se forja a través de dichos materiales y flujo de energía vital que usualmente es obtenido por herreros en el proceso de forja, permite a los usuarios desarrollar mejoras mágicas, así como físicas.

## **Manera del juego**

Las **colisiones** son una parte fundamental del juego pues nos permiten mostrar interacciones entre los objetos que son creados dentro el nivel, cada figura tiene su propia área o a lo que denominamos coordenada en X y Y, estas nos permiten saber en qué área del mapa se encuentra cierto objeto, y es con la cual damos instancia a diversos, las colisiones nos permiten interactuar a los 2 personajes principales que son los magos Aria y Hygei con los objetos que debe de pasar, los que son obstáculos y enemigos, así como los tesoros que encontraran.

Para hacerlo resumido, cada colisión se va a dar cuando nuestros magos interactúen o en su defecto colisionen con otro objeto del juego, en este caso, damos por declaración que cuando nuestro objeto se mueva sus coordenadas de X y Y se estarán incrementando o decrementado, pero los objetos del juego ya están por defecto en una área, una coordenada en concreto, y es aquí donde repartimos en caso que si los magos llegan a cierta coordenada, estarán colisionando con los objetos, y por tanto deberán hacer ciertas acciones.

Para los enemigos solamente comparamos sus variables, ya que estos están en constante movimiento, y utilizamos hilos para que realiza en sus movimientos, es así como cuando los magos y enemigos en sus variables de coordenadas lleguen a ser las mismas, significa que estos se habrán encontrado y por tanto la colisión estará hecha

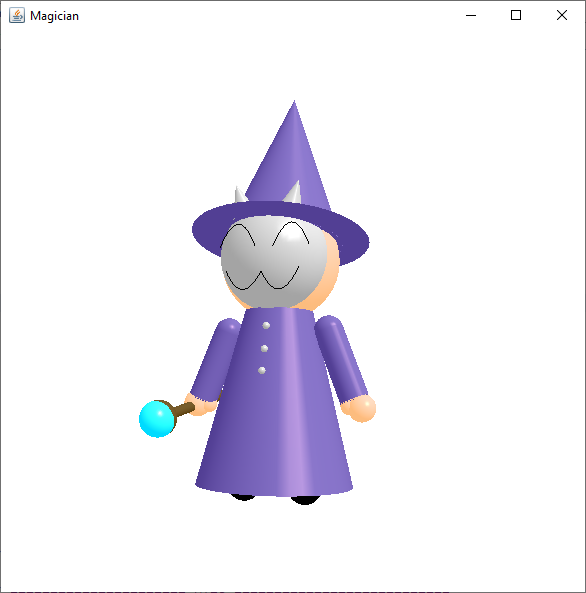
La **cámara** en el juego nos permite esa inmersión de que nuestro personaje esta interactuando con los mapas, nuestro escenario es en 3D y las figuras son en 3D, por tanto, hicimos la cámara estática que solamente se pueda mover de izquierda y derecha cuando el personaje ejecute las acciones de movimiento de izquierda a derecha.

En resumen, la cámara solo esta estática y se mueve de izquierda a derecha y arriba hacia abajo cuando realiza saltos y caídas el mago.

EL objetivo principal del juego es permitir llegar a los magos a su destino designado con una bandera y en lo largo de estos evitar los peligros, tomara los ítems dispersos en los mapas y pasara los obstáculos.

## Personajes 3D

**Hygel**

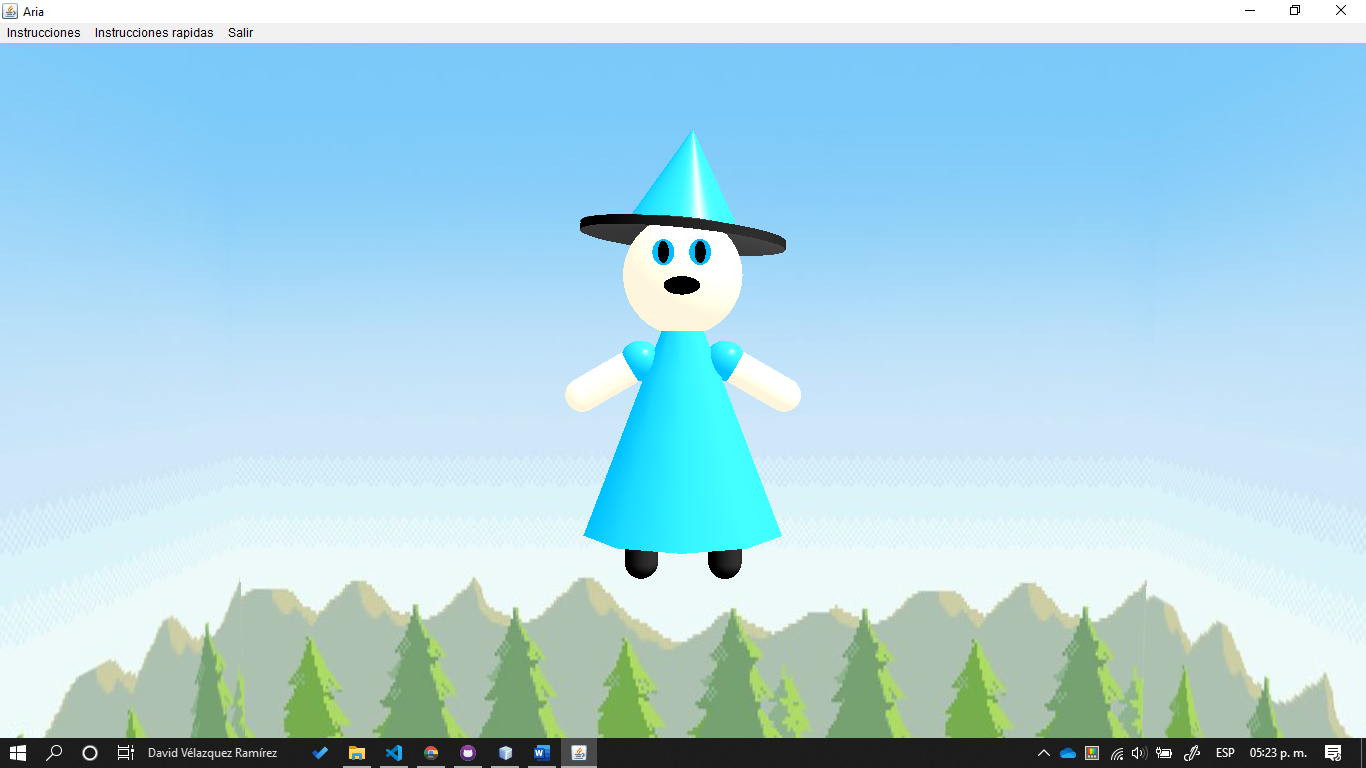


(Carlos López)

2. Mago Soporte

Pertenece al grupo FRAGMA, desarrollo sus habilidades en la gran biblioteca de AERIX donde adquirió conocimiento sobre hechizos, debido a que posee un KEYLUK basado en tierra, potencia sus habilidades de forma incomparable y genera un gran escudo el cual lo protege de grandes daños.

**Aria**



(David Velázquez)

3. Aria

Pertenece al grupo FRAGMA, sus mejores habilidades destacan en movimientos físicos avanzados, con su KEYLUK basado en agua adquiere mayor agilidad y destreza a la hora de ejecutar sus movimientos.

**Flogats**

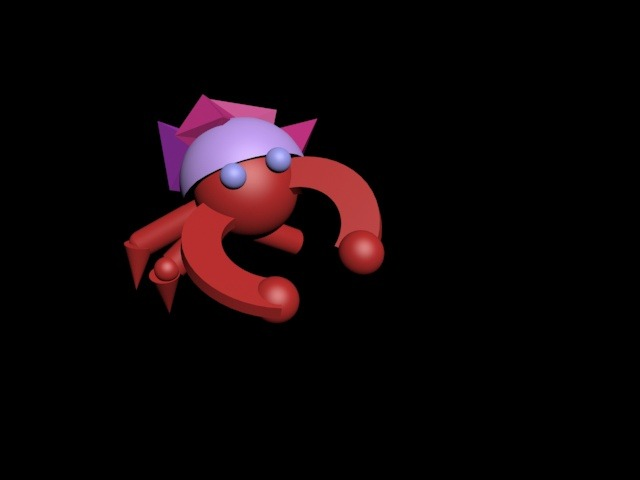


(Areli Lara)

4. Flogats

Monstruo formado de la esencia de la luktita que se emite de la mina, su apariencia es de fuego y asimila a una llama, se encuentra de distintos colores dependiendo el ambiente en el que se encuentre.

**Terluks**



(Alan Velázquez)

5. Terluks

Monstruos formados con el paso de los años a través del mineral luktita, protegen el mineral a toda costa, un peligro al cual se debe enfrentar el grupo FRAGMA en su travesía por el valioso ítem.

### **Análisis de requerimientos**

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF1** | |
| **Nombre Del Requerimiento** | Movimientos del personaje | |
| **Características** | El personaje seleccionado puede realizar movimientos. | |
| **Descripción** | El personaje seleccionado puede realizar los siguientes movimientos, y cada movimiento realiza una expresión corporal o facial distinta:   * Saltar: cerrar los ojos * Caminar a la derecha: mover los brazos * Caminar a la izquierda: mover los brazos * (Caída: este movimiento es resultado de saltar y regresar al suelo. Mantendrá la expresión del salto si el personaje no realiza otra acción.) * El personaje tambien tiene colisiones con otros elementos del nivel seleccionado, como obstáculos que le impidan moverse o enemigos que al contacto con ellos termine el juego. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF2 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF** | |
| **Nombre Del Requerimiento** | Lectura del teclado | |
| **Características** | El usuario puede utilizar las teclas para mover al personaje. | |
| **Descripción** | El usuario podrá mover a su personaje seleccionado presionando las teclas de movimientos predefinidas:  W – Saltar  A – Caminar izquierda  D – Caminar derecha  Durante la caída del salto, el usuario puede moverse a los lados para mejor manejabilidad del personaje. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF1 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF3** | |
| **Nombre Del Requerimiento** | Seleccionar nivel | |
| **Características** | Permite al usuario seleccionar run nivel para jugar. | |
| **Descripción** | El usuario puede seleccionar un nivel de los disponibles utilizando su *mouse* y haciendo clic sobre la imagen del nivel:   * 4 niveles disponibles (enumerados respectivamente)   Al seleccionar un nivel, el color de la imagen del nivel cambia de color, para mostrar que esta seleccionado.  El botón con una flecha pasará a la siguiente ventana de selección de personaje.  Si no se ha seleccionado un nivel, antes de presionar la flecha, la ventana no cambiará a la de selección de personaje y hará un sonido de “bloqueo”. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF4 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF4** | |
| **Nombre Del Requerimiento** | Selección de personaje | |
| **Características** | El usuario puede seleccionar un personaje para jugar, utilizando su *mouse.* | |
| **Descripción** | * 2 personajes seleccionables disponibles.   El usuario podrá hacer clic con su *mouse* sobre el personaje que desea jugar.  Los personajes se encuentran rodados de un marco color azul. Al posar el *puntero* sobre el personaje si hacer clic, este cambiara a color amarillo; si retira el *puntero* del personaje, este volverá a color azul.  Al hacer clic sobre el personaje, el marco que lo rodea cambiara de color dorado, y aunque vuelva a posar el *puntero*sobre el personaje, este no cambiara a menos que se haga *clic*.  Si ya ha sido seleccionado un personaje, y se hace *clic* en otro el seleccionado previamente volverá a su color original azul y el nuevo personaje seleccionado se pondrá en color dorado.  En la parte inferior derecha se encuentra el botón inicio para comenzar el juego. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF3 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF5** | |
| **Nombre Del Requerimiento** | Cargar nivel | |
| **Características** | Carga el nivel con las características seleccionados por el usuario (nivel y personaje). | |
| **Descripción** | El nivel seleccionado por el usuario se muestra en la pantalla junto con el personaje seleccionado.  Los 4 niveles son distintos entre si y contienen diferentes elementos como:   * Obstáculos * Enemigos * Objetos   El usuario puede viajar a través del nivel, y tendrá que llegar hasta la zona de meta para ganar. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF3, RF4 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF6** | |
| **Nombre Del Requerimiento** | Pausar nivel | |
| **Características** | El usuario puede poner pausa al juego durante la partida. | |
| **Descripción** | El usuario podrá poner en pausa su partida al presionar la tecla *ESC* de su teclado.  Al ser presionada mostrara una ventana con la opción activar o desactivar el volumen, y las teclas de movimientos.  Para quitar la pausa, será necesario presionar nuevamente la tecla *ESC.* | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF5 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF7** | |
| **Nombre Del Requerimiento** | Configurar | |
| **Características** | Muestra la opción de quitar la música.  Muestra las teclas de los movimientos de personajes. | |
| **Descripción** | La pantalla muestra el mensaje “Música de fondo” y las opciones “ON” y “OFF”, para activar o desactivar la música, respectivamente.  Debajo muestra el mensaje “Movimientos” y muestra los movimientos que realiza cada tecla.   * Saltar: cerrar los ojos * Caminar a la derecha: mover los brazos * Caminar a la izquierda: mover los brazos   En la parte inferior derecha está el botón inicio, para regresar a la pantalla principal. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF1, RF2 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF8** | |
| **Nombre Del Requerimiento** | Créditos | |
| **Características** |  | |
| **Descripción** | Muestra la información de los desarrolladores.  En la parte inferior derecha está el botón inicio, para regresar a la pantalla principal. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF5 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RF8** | |
| **Nombre Del Requerimiento** | Movimiento de cámara | |
| **Características** |  | |
| **Descripción** | La cámara se centra en el personaje, y se mueve con el personaje. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF1 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RNF1** | |
| **Nombre Del Requerimiento** | Buen sonido | |
| **Características** | El sonido es audible. | |
| **Descripción** | El audio de fondo es de buena calidad, puede escuchar se forma simple y puede ser desactivado si el usuario lo solicita. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RF7 |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RNF2** | |
| **Nombre Del Requerimiento** | Colores agradables | |
| **Características** | Los colores no son molestos para la vista del usuario. | |
| **Descripción** | Los colores de los escenarios, botones, personajes y demás, no presentan molestias a la vista del usuario, son visibles y concuerdan con los elementos del juego. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Media | | Exigible |
| **Requerimientos relacionados** | |  |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RNF3** | |
| **Nombre Del Requerimiento** | Historia | |
| **Características** |  | |
| **Descripción** | La historia es fácil de entender y entretenida para el usuario. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Media | | Exigible |
| **Requerimientos relacionados** | |  |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RNF3** | |
| **Nombre Del Requerimiento** | Buena resolución | |
| **Características** | La resolución del juego es buena para el usuario y para cualquier equipo de cómputo. | |
| **Descripción** | La ventana del juego tiene las medidas de 820px X 573px.  Esto para evitar que pantallas con poca resolución no pendan manejar bien la vista, así como para que no resulte moleste para el usuario y pueda enfocarse en toda el área de forma fácil. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | |  |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RNF4** | |
| **Nombre Del Requerimiento** | Textos visibles | |
| **Características** | Los mensajes son visibles para el usuario. | |
| **Descripción** | Los mensajes, botones y letreros del juego están escritos con la fuente Tahoma Plain, con un tamaño de 18px | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | |  |

|  |  |  |
| --- | --- | --- |
| **Nombre del Proyecto** | **Aerix** | |
| **Identificación De**  **Requerimientos** | **RNF5** | |
| **Nombre Del Requerimiento** | Sin lag | |
| **Características** |  | |
| **Descripción** | El juego funcionará en la mayoría de equipos con Windows, pero la función del juego, dependerá del equipo en que se este ejecutando, por lo que algunos equipos de bajo rendimiento tendrán problemas de *FPS*. | |
| **Actores** | Usuario | |
| **Prioridad de Requerimientos** | | **Importancia de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | |  |

### **Casos de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **RF1** | | Movimientos del personaje | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | YouTube, OpenGL for Dummys | |
| **Objetivos asociados** | | Movimientos de los personajes | |
| **Descripción** | | El personaje se mueve cuándo se presiona alguna tecla de movimiento | |
| **Precondición** | | Debe ser presionada la tecla que realiza tal acción del personaje. | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **Usuario** | Mueve al personaje en la dirección que tiene establecida la tecla | |
| **2** | **Usuario** | W para que el personaje salte | |
| **3** | **Usuario** | D para que el personaje avance a la derecha | |
| **4** | **Usuario** | D para que el personaje avance a la izquierda | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** | **Usuario** | **Puede realizarse un salto, y durante la caída realizar otro movimiento.** | |
| **2.1** | **Mono y obstáculo** | **No se puede avanzar si hay un obstáculo frente al personaje** | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Caminar derecha** | | 1 | 1 segundo |
| **Caminar izquierda** | | 1 | 1 segundo |
| **Saltar** | | 1 | 3 segundos |
| **Frecuencia esperada** | | 1 / segundo | |
| **Importancia** | | Vital | |
| **Postcondición** | | Cambio de posición del mono | |
| **Urgencia** | | inmediatamente | |
| **Comentarios** | | Problemas con los obstáculos y movimientos múltiples. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RF2** | | Lectura de teclado | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | Youtube, OpenGL for Dummys | |
| **Objetivos asociados** | | Leer el de teclado para mover al personaje | |
| **Descripción** | | Al presionar una tecla se realiza una acción de movimiento o carga de pantalla. | |
| **Precondición** | | Debe presionarse una tecla | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **W** | El personaje salta | |
| **2** | **A** | El personaje se mueve a la derecha | |
| **3** | **D** | El personaje se mueve a la izquierda | |
| **4** | **ESC** | Se pausa el juego y se muestra la pantalla de pausa | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** | **Usuario** | **Al presionar la tecla w para saltar y otra tecla de movimiento (A o D) se realizan dos acciones al mismo tiempo** | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Presionar W** | | 1 | 1 segundo |
| **Presionar D** | | 1 | 1 segundo |
| **Presionar A** | | 1 | 3 segundos |
| **Presionar ESC** | | 1 | 1 segundo |
| **Frecuencia esperada** | | 1 / segundo | |
| **Importancia** | | Vital | |
| **Postcondición** | | Cambio de posición del mono | |
| **Urgencia** | | inmediatamente | |
| **Comentarios** | | Problemas con los obstáculos y movimientos múltiples. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RF3** | | Seleccionar nivel | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | YouTube, OpenGL for Dummys | |
| **Objetivos asociados** | | Seleccionar un nivel de los 4 disponibles. | |
| **Descripción** | | Utilizar el mouse para seleccionar un nivel, al darle clic encimade la imagen del nivel. | |
| **Precondición** | | Hacer clic en la imagen del nivel | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **Usuario** | El nivel sobre el que se hizo clic cambia el color de su imagen | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** | **Usuario** | **Si un nivel ya ha sido seleccionado y se selecciona otro, es anterior regresa a su color inicial y el segundo** | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Clic en nivel** | | 1 | 1 segundos |
| **Clic en nivel con otro seleccionado** | | 2 | 1 segundos |
| **Frecuencia esperada** | | 3 veces / minuto | |
| **Importancia** | | Vital | |
| **Postcondición** | | Cambio de imagen | |
| **Urgencia** | | inmediatamente | |
| **Comentarios** | | Azul es un color | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RF4** | | Seleccionar personaje | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | YouTube, OpenGL for Dummys | |
| **Objetivos asociados** | | Selecciona un personaje | |
| **Descripción** | | Se selecciona un personaje al hacer clic sobre el | |
| **Precondición** | | Debe cargarse la animación de los personajes seleccionables | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **Usuario** | Cambia el color del marco que contiene al personaje | |
| **2** | **Usuario** | Posiciona el puntero en el marco y este cambia de color momentáneamente | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** | **Sistema** | **Si ya se había seleccionado un personaje y se selecciona otro, es primero regresa a su color original-** | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Seleccionar personaje** | | 1 | 1 segundos |
| **Iniciar** | | 2 | 1 segundos |
| **Frecuencia esperada** | | 3/minuto | |
| **Importancia** | | Vital | |
| **Postcondición** | | Cambio de imagen | |
| **Comentarios** | |  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RF5** | | Configurar | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | Youtube, OpenGL for Dummys | |
| **Objetivos asociados** | | Modificar audio y muestra teclas de movimiento | |
| **Descripción** | | Permite modificar la música de fondo, encender o apagarla y muestra las teclas de movimiento | |
| **Precondición** | | Abrir la ventana de configuración | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **Usuario** | Hacer clic en “ON”, prende el audio | |
| **2** | **Usuario** | Hacer clic en “OFF”, apaga el audio | |
| **3** | **Usuario** | Hacer clic en “inicio”, muestra la pantalla principal | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** |  |  | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Clic en encender audio** | | 1 | 1 segundos |
| **Clic en apagar audio** | | 2 | n segundos |
| **Frecuencia esperada** | | 1 / hora | |
| **Importancia** | | importante | |
| **Postcondición** | | modifica el audio | |
| **Urgencia** | | {hay presión | |
| **Comentarios** | |  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RF6** | | Créditos | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | Youtube, OpenGL for Dummys | |
| **Objetivos asociados** | | Muestra créditos | |
| **Descripción** | | Muestra información de desarrolladores | |
| **Precondición** | |  | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **Usuario** | Se abre al hacer clic en la opción créditos. | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** |  |  | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Clic en créditos** | | 1 | 1 segundo |
| **Clic en inicio** | | 2 | 1 segundo |
| **Frecuencia esperada** | | 1 / hora | |
| **Importancia** | | importante | |
| **Postcondición** | |  | |
| **Urgencia** | | no hay presión | |
| **Comentarios** | | ici pisi | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RF7** | | Cargar nivel | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | Youtube, OpenGL for Dummys | |
| **Objetivos asociados** | | Carga el nivel seleccionado | |
| **Descripción** | | Se carga el nivel y el personaje seleccionados. | |
| **Precondición** | | Debe tenerse seleccionado un nivel y un personaje | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **Usuario** | Se carga el nivel, personaje, enemigos y objetos | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** |  | **No se puede acceder a esta pantalla si no se selecciona un nivel y un personaje.** | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Clic en encender audio** | | 1 | segundos |
| **Clic en apagar audio** | | 2 | segundos |
| **Frecuencia esperada** | | 1 / hora | |
| **Importancia** | | importante | |
| **Postcondición** | | Muestra el nivel | |
| **Urgencia** | | hay presión | |
| **Comentarios** | | me quiero morir | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RF8** | | Pausar nivel | |
| **Versión** | | 3.0 | |
| **Autores** | | Carlos López, David Velázquez, Alan, Areli | |
| **Fuentes** | | Youtube, OpenGL for Dummys | |
| **Objetivos asociados** | | Pausar la partida | |
| **Descripción** | | Se pausa el juego y muestra la pantalla de pausa, con las teclas y la opción de audio. | |
| **Precondición** | | Presionar la tecla ESC | |
| **Secuencia Normal** | |  | |
|  | **Actor** | **Sistema** | |
| **1** | **Usuario** | Se muestra la pantalla de pausa al presionar la tecla ESC | |
| **Excepciones** | |  | |
|  | **Actor** | **Sistema** | |
| **1.1** |  |  | |
| **Rendimiento** | | **Paso** | **Cota de tiempo** |
| **Pulsar ESC para pausa** | | 1 | segundos |
| **Quitar pausa** | | 2 | segundos |
| **Frecuencia esperada** | | 6 / hora | |
| **Importancia** | | importante | |
| **Postcondición** | | Pausa la partida | |
| **Urgencia** | | hay presión | |
| **Comentarios** | | ¿A quién se le ocurrió esto? | |

# Bibliografía

OpenGL programming guide: the official guide to learning OpenGL, version 4.3 / Dave Shreiner, Graham Sellers, John Kessenich, Bill Licea-Kane; the Khronos OpenGL ARB Working Group. ---Eighth edition. Person editorial.

Boreskov, Alexey, Shikin, Evgeniy. (2013). Computer Graphic: From Pixels to programmable Graphics Hardware. CRC Press.

Huges, John F. (2013). Computer Graphics: principles and practice. 3rd Edition. Pearson Education, Inc.

Xiang, Zhigang, Plastock Roy. Computer Graphics. 2nd edition. Mc Graw Hill, Shaum´s Outlines. USA, 2000.

# Anexos

<https://learnopengl.com/book/offline%20learnopengl.pdf>