



ESCUELA
NACIONAL
DE ESTUDIOS
SUPERIORES

UNIDAD MORELIA

Administración de proyectos

Profesor: Héctor Alonso Guzmán Gutiérrez

Plan de Proyecto

ENESoftware

Jorge Antonio Camarena Pliego
David Calderón Ceja
Keshava Tonathiu Sánchez Barbosa
Stephany Dzoara Vargas Mier

13/09/2019

Semestre 2020-1

Versión 0.2

Índice

Enunciado del problema	3
Descripción	3
Características	3
Arquitectura	3
Actividades de desarrollo	4
Recursos	4
Comunicación en el equipo	4
Comunicación con el cliente	4
Riesgos	5
Administración de la configuración	5
Flujos de Trabajo	6

Enunciado del problema

Descripción

El producto que se desarrollará será un videojuego de género “platformer” (o “plataformero” en español) 2D, donde el concepto principal es un personaje que se mueve a lo largo del eje x en dirección a los positivos hasta que llega a un punto objetivo o “checkpoint” donde se considera el fin del escenario.

El juego tendrá múltiples escenarios y el objetivo será llegar al final de todos los escenarios, con una pequeña trama que consiste en que el protagonista pretende salvar al mundo del silencio, revolviéndose los niveles en torno a música única.

Características

1. El juego será creado para ser ejecutado en computadoras, tendrá un “aspect ratio” de 16:9, la resolución del juego será de 1280x720 píxeles.
2. La temática visual principal del juego serán luces neon.
3. El jugador podrá brincar, agacharse y moverse derecha e izquierda.
4. Cada nivel del juego tendrá un género de música diferente.
5. Al final de cada nivel habrá una sección particularmente difícil de superar por parte del jugador en comparación al resto del nivel.
6. Cada nivel es progresivamente más difícil y requerirá más destreza y rapidez posiblemente por parte del jugador.

Arquitectura

El juego será desarrollado con una arquitectura de 32 bits, de modo que la computadora en la que se quiera instalar deberá tener un sistema operativo Windows con dicha condición, sin garantía de que se produzca alguna versión estable para Linux o MacOS.

En cuanto a los requerimientos de hardware mínimos para ejecutar el juego sin problemas, se deben tener estas especificaciones:

- CPU: Intel Core 2 Duo
- RAM: 2GB
- GPU: Intel HD Graphics 2000 / Nvidia GeForce 8600 GT / Radeon HD 2600
- Espacio Libre en Disco: 150 MB

En cuanto a la distribución o disponibilidad del producto final, se le ofrecerá al cliente un enlace de descarga con el instalador del juego.

Casos de Uso

Dada la naturaleza del producto en cuestión, los casos de uso son recreamiento y ocio.

Actividades de desarrollo

El proyecto se desarrollará a lo largo del semestre 2020-1, comenzando el **5 de septiembre de 2019** y finalizando el **21 de noviembre de 2019**.

Fechas tentativas de entrega(sujetas a cambios):

15/Octubre/2019- entrega de diseño.

24/Octubre/2019- Git.

29/Octubre/2019- Prototipo.

14/Noviembre/2019- Entrega final.

Recursos

En cuanto a recursos humanos, tendremos a nuestra disposición la totalidad de los cuatro integrantes del equipo ENESoftware, con la potencial colaboración de una estudiante de la carrera de Arte y Diseño de la Escuela Nacional de Estudios Superiores Morelia, UNAM para el diseño de algunas características gráficas y artísticas.

Comunicación en el equipo

La comunicación se efectuará principalmente a través del servicio de mensajería WhatsApp.

Comunicación con el cliente

La comunicación con el cliente se podrá abordar a través del mismo servicio gratuito de github que permite manejar “issues” de retroalimentación y discusiones a manera de foro.

Riesgos

Prob. de impacto	Bajo	Medio	Alto
0.8	★ Enfermedad de algún integrante del equipo	★ Problemas con uso de software desconocido	★ Mala comunicación en cuestiones de diseño e implementación
0.5	★ Deserción de algún integrante del equipo	★ No hay suficiente personal para la realización del proyecto	★ Retraso en actividades de desarrollo
0.25	★ Falta de equipo y software necesario para trabajar	★ Cambios imprevistos del proyecto	★ Robo/pérdida de información y trabajo

Administración de la configuración

Para el manejo versionado de todo el proyecto, tomando en cuenta tanto los documentos formales de especificaciones como los "assets" o contenido multimedia del juego así como su código y otros elementos misceláneos para compilar en Unity el proyecto, trabajaremos directorios en un repositorio de GitHub junto con su respectivo ReadMe en el que se incluyan instrucciones respecto a cómo añadir el proyecto a Unity, así como la especificación de la versión de Unity trabajada en el proyecto para evitar problemas de incompatibilidad.

Directorios:

UnityProject: Aquí estarán todos los elementos necesarios para agregar y trabajar todos los elementos fuente del proyecto en Unity, tanto código fuente como assets.

Documentation: En este directorio incluimos todos nuestros documentos de especificaciones respecto a la planeación del proyecto, es decir, la documentación

de todo el proyecto. Principalmente la parte de requerimientos, análisis y diseño en la metodología seguida. Se incluirán tanto en formato editable .docx como en formato .pdf formal. Estos documentos serán los siguientes:

- **FormatoDeLaDocumentacion:** En este documento se especifica el formato que tendrán todos los documentos dentro de esta carpeta. Tanto la fuente de la letra usada, como su tamaño, su interlineado, tamaño de los saltos de línea, así como formato de la portada y de las páginas en general, como su margen.
- **PlanDeProyecto:** En este documento, que es el actual tratado, se enuncia a muy grandes rasgos los requerimientos, así como se especifica la forma en que se plantea trabajar el proyecto entre el equipo y con el cliente.
- **Requerimientos:** En este documento se abarca de la forma más exhaustiva posible todo lo que debe ser capaz de cumplir el producto final de este proyecto. Tanto requerimientos funcionales que le permiten al juego funcionar, como los no funcionales que le dan calidad al producto
- **Arquitectura:** En este documento relativamente breve por no requerir una arquitectura compleja para la realización del proyecto, se especifica con diagramas las relaciones y propiedades de los objetos en el proyecto trabajado de Unity. Por tratarse de un motor de juego cuyo paradigma es conocido como Entity-Component, basado precisamente en que cada entidad posee un listado de componentes reusables y globales, no se tratará un diagrama de clases exhaustivo como tal, pero similar a él haciendo énfasis en dichos componentes que conforman las propiedades de las entidades GameObject.

El desarrollo del juego será completamente llevado a cabo en la plataforma Unity, desarrollando en el lenguaje de programación “C#” (Csharp).

Gran parte del trabajo será realizado en tareas individuales por cada integrante del equipo, así que será necesaria una plataforma para compartir todos los recursos y configuraciones del proyecto en línea. En este caso, optamos por Github gracias a no tener un costo implicado, a diferencia del servicio que ofrecen los mismos desarrolladores de Unity: Unity Teams, que solo es gratuito hasta un máximo de tres miembros por equipo y al ser cuatro nosotros, no sería ideal. El único inconveniente de github, si se quiere ver así, será que nuestro trabajo va a ser completamente libre y público con licencia GNU.

Flujos de Trabajo

Estos serán los flujos de trabajo en que dedicaremos más tiempo de trabajo como equipo.

- **Dirección artística:** Al tratar un producto naturalmente multimedia, necesitamos trabajar mucho la sección artística del proyecto, estando en muy cercano contacto con nuestro contacto de la carrera de Arte y Diseño y en

caso necesario tener que trabajar nosotros como parte principal del equipo generar o conseguir el resto de los assets necesarios.

- Implementación: Claramente será necesaria una buena cantidad de trabajo de bajo nivel para que todos nuestros requerimientos se vean satisfechos, especialmente al trabajar con los scripts de C# y que el manejo adecuado de los componentes de todos los objetos trabajados.
- Pruebas: Siendo que se trata de un juego de físicas, así como de un producto que puede llegar a ser relativamente pesado computacionalmente, debemos garantizar que no tendremos fenómenos indeseables como "overflows" de memoria ni lentitud frustrante para el cliente en una experiencia que debería ser divertida.