

Estudiantes:

Daniel Ramos Guerrero & Gabriel Murillo López

1. Atributo de Análisis de Problema

Con este proyecto se presenta un problema de ingeniería interesante, el cual consta de crear un videojuego que representa cierto grado de complejidad, dado a sus requerimientos. Entre los cuales se encuentra la generación de un mapa aleatorio pero válido, “la inteligencia artificial” para que los enemigos de este videojuego puedan moverse, entre otros. Estos problemas, dado que son de computación, y la computación es intrínsecamente matemática, se pueden representar con grafos o árboles.

En términos de desarrollo sostenible, podemos ver cómo trabajar en este proyecto y desarrollar una solución se alinea con el objetivo número 4 de desarrollo sostenible de la ONU, que es la educación de calidad. En el transcurso del desarrollo de este, se desarrollan habilidades de programación e investigación que serán de gran ayuda en el mercado laboral.

Para llegar a una solución para este problema, se puede aplicar la técnica de “divide y vencerás”; en vez de ver el juego como una sola cosa, se pueden subdividir sus partes en tareas más pequeñas que se pueden hacer con mayor facilidad. Por ejemplo, el juego se puede dividir en los siguientes aspectos: Creación de sprites, generación de mapa, desarrollo del personaje principal, desarrollo del enemigo, desarrollo del GUI, etc. Estas subtareas se pueden volver a dividir. La generación de mapa se puede subdividir en implementación de clases de terreno, implementación de BFS para encontrar un camino válido. Desarrollo de GUI en implementación de pantalla de puntuaciones, pantalla de ajustes, etc.

El pro de hacer esta división es que nos da tareas pequeñas más administrables; además, se facilita la colaboración en grupo. Pero en contra, puede ser que al tener todas estas subtareas y subsubtareas sea difícil escoger dónde empezar. Un proyecto como este nos permite aprender no solo del área de programación, sino también de matemática e incluso artística.

2. Atributo de Herramientas de Ingeniería

Durante la realización de este proyecto se utilizan múltiples herramientas. GitHub se utiliza como un sistema de control de versiones que, además de ayudar a mantener respaldos del código, facilita la colaboración, y utilizando funciones como los “issues” de un repositorio. Otra herramienta muy importante durante este proyecto es un editor de código que ayude a agilizar el desarrollo, VS Code.

Como se mencionó anteriormente, la técnica principal para realizar este proyecto es la subdivisión en problemas más pequeños. Ya con estos subproblemas se puede investigar cómo implementar ciertos algoritmos, cómo hacer sprites, etc. GitHub también ayuda en este paso, conteniendo una cantidad amplia de código de referencia.

La puntuación funciona de esta manera: Dependiendo de la dificultad seleccionada, se aplicará un multiplicador, sin importar el modo. x1 para fácil, x1.5 para normal y x2 para difícil.

En modo Escapa, la puntuación se calcula aplicando el multiplicador de dificultad, sumando 50 por cada enemigo que se atrape y dependiendo del tiempo que se tarda en escapar: si es menos de 5 segundos, 1000 puntos, y ahí en intervalos de 5 segundos se va reduciendo por 100.

En modo Cazador se aplica el multiplicador, se suma +200 por cada enemigo atrapado y se resta -100 por cada enemigo escapado.

A continuación, se encuentra el diagrama de clases; acá se grafican las clases que se utilizaron y cómo se relacionan.

