

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS
INSTITUTO DE INVESTIGACIÓN EN CIENCIAS BÁSICAS Y APLICADAS
CENTRO DE INVESTIGACIÓN EN CIENCIAS**

MS-2k1

REPORTE

**INTRODUCCIÓN A LA PROGRAMACIÓN DE VIDEOJUEGOS EN
C++ Y PYTHON**

PRESENTA

***ANDRÉS REYES - FRANCISCO AQUINO -
EDUARDO NUÑEZ***

PROFESOR:

Mtro. Yainier Labrada Nueva

Índice general

1. Introducción	2
2. Objetivos	3
2.1. Objetivos específicos	3
2.2. Objetivos generales	3
3. Marco teórico	4
3.1. Anaconda [2]	4
3.2. Python [3]	4
3.3. Pygame [4]	5
3.3.1. Sprites	5
4. Composición y desarrollo	6
4.1. Nociones básicas del juego desarrollado	6
4.2. Partes integrales del juego	6
4.2.1. Pantalla	6
4.2.2. Naves	7
5. Conclusiones	9

Capítulo 1

Introducción

En los últimos años, la industria de los videojuegos se ha visto envuelta en una revolución tecnológica abismal, el poder de computo y las ascendentes tecnologías en el área de la comunicación digital, han abierto puertas inmensas para todas las ramas de tecnología en el mundo, entre ellas, el desarrollo de videojuegos.

Los videojuegos recrean entornos y situaciones virtuales en los que la persona puede controlar a uno o mas personajes de este entorno.

Este proyecto tiene el enfoque de poder aprender y analizar la programación de videojuegos, que emplea muchos paradigmas de la programación como la programación orientada a objetos, a eventos, procesos, y muchos conceptos más. [1]

1.1. Videojuegos

Los videojuegos, como concepto, se componen por distintas partes, una de ellas es el controlador, que es un dispositivo de entrada que varía dependiendo de la plataforma. Por ejemplo, un controlador podría consistir únicamente de un botón y una palanca, mientras que otros podrían tener más de una docena de botones y una o más palancas.

Capítulo 2

Objetivos

Veni vidi vici.

2.1. Objetivos específicos

Esta sección esta enfocada a las distintas metas que se propusieron alcanzar como base para la realización de este proyecto. Las siguientes son:

- Adquirir las bases y el conocimiento para poder desarrollar un videojuego desde cero.
- Comprender como se emplea la programación y la lógica en algo tan complejo como lo es un videojuego.
- Entender la modelación y aplicación de la programación orientada a objetos en un sistema en tiempo real, con eventos y objetos en constante cambio, dentro de un entorno simulado.

2.2. Objetivos generales

Los objetivos enlistados aquí, son metas que sobre la marcha, se vislumbraban completables a lo largo del desarrollo del videojuego. Algunas de ellas son:

- Aprender y reforzar un lenguaje de programación de alta popularidad como lo es Python [3].
- Aplicar y desarrollar conceptos adquiridos en la formación académica en un proyecto real e integral.
- Desarrollar y pulir las distintas habilidades de programación que se han adquirido a lo largo de la formación académica.

Capítulo 3

Marco teórico

El marco teórico que presentamos aquí basado en una exhaustiva investigación para poder corroborar cuales eran los mejores métodos para implementar el proyecto.

3.1. Anaconda [2]

Anaconda es una distribución libre y abierta de los lenguajes Python y R. Utilizamos anaconda por su flexibilidad y por su completud en cuanto a entornos concierne, el ecosistema de Python permite hacer instalación de paquetes de una forma muy sencilla y rápida.

Dentro de Anaconda utilizamos JupyterLab, que es una herramienta que nos permite visualizar y editar el código por celdas, manteniendo un mejor control de los errores.

3.2. Python [3]

Python es un lenguaje de programación de alto nivel, muy sencillo de aprender a utilizar, y además es multiparadigma, por lo que Python se presta para realizar cualquier tipo de tarea, como es el caso: un videojuego.

Es un lenguaje de programación versátil, utilizado en muchas áreas, como lo son la estadística, la comunicación y la ciencia de datos.

Algunas de sus ventajas son:

- Es simple y rapido: puedes hacer mucho con pocas lineas de código.
- Flexible: No necesariamente vas a tener que preocuparte por detalles de tipificación y declaraciones.
- Multiplataforma: Se puede ejecutar en cualquier sistema operativo y cuenta con librerías específicas para cada uno de estos.

-
- Open Source: Su comunidad se encarga de mantener al lenguaje y crear todos sus recursos de forma gratuita.

3.3. Pygame [4]

Pygame es un conjunto de módulos del lenguaje Python que permiten y facilitan la creación de videojuegos en dos dimensiones de una manera sencilla, gracias al manejo de sprites.

Este conjunto de modulos, permite una flexibilidad y manejo en la manipulación de sprites.

3.3.1. Sprites

Los sprites sirven para la optimización del uso de imágenes empleadas en CSS. Mediante la construcción de una cuadrícula de imagenes ordenadas y el uso de estilos, se pueden ubicar todas las imagenes mostrando solo una porción de la imagen. Esto optimiza tiempos de descarga de una página como rendimiento de las mismas.



Figura 3.1: Ejemplo de sprites

Capítulo 4

Composición y desarrollo

Augere.

4.1. Nociones básicas del juego desarrollado

El presente proyecto es un videojuego desarrollado en el lenguaje de programación Python, haciendo uso de los módulos de Pygame; el cual esta diseñado para dos jugadores simultáneos los cuales controlan dos naves espaciales en un enfrentamiento y gana aquel jugador que terminé primero con las vidas del oponente.

4.2. Partes integrales del juego

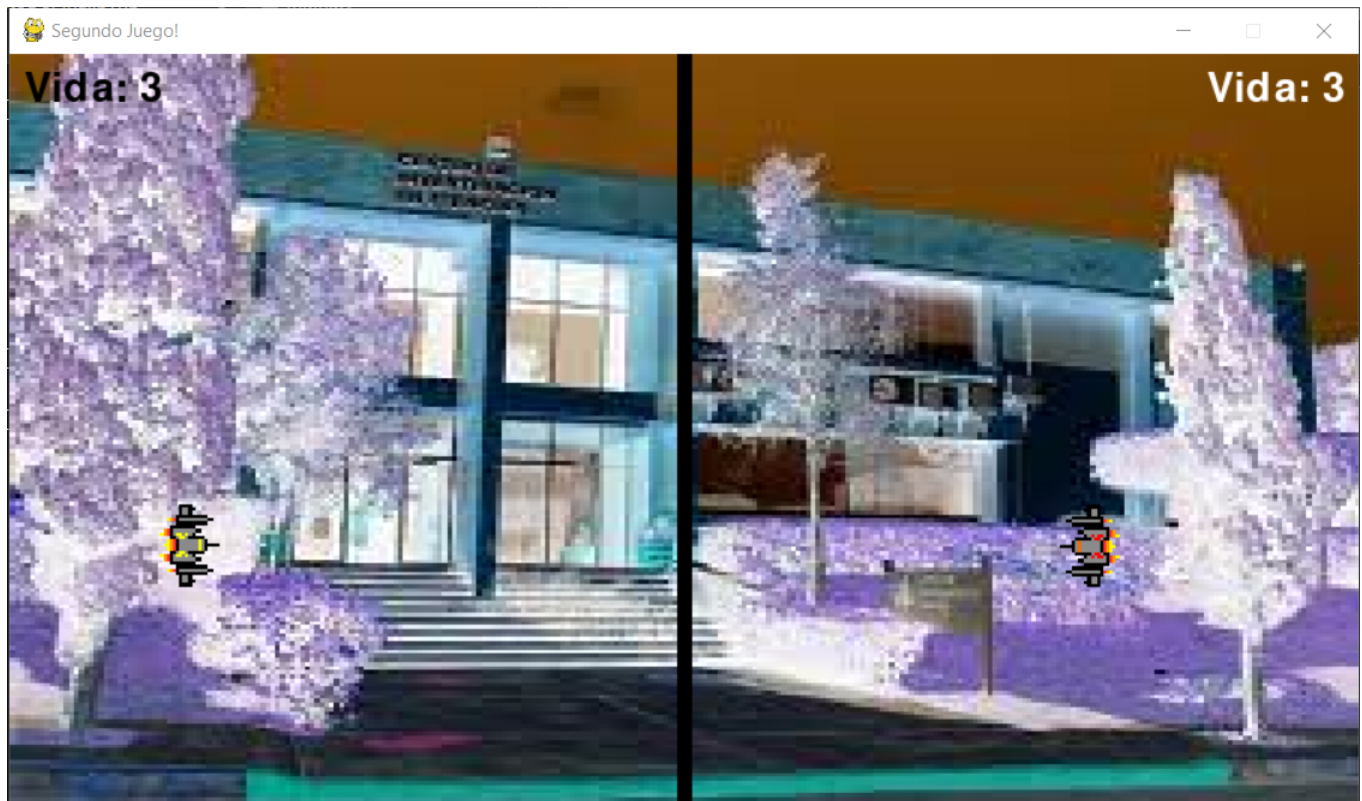
El videojuego se compone de tres partes principales, las cuales son:

4.2.1. Pantalla

La pantalla principal del juego es un fondo estático de dimensiones 900x500 pixeles, sobre el cual se dibujan los demás componentes del juego. Como imagen de fondo se uso una foto del Centro de Investigación en Ciencias convertida a negativo, como se muestra a continuación:



Dicha imagen se divide a la mitad con el propósito de asignar a cada jugador ese espacio que actúa como su área de juego desde el cual ninguno de los dos se puede mover fuera de la misma. En las esquinas superiores se dibujan a manera de texto las vidas restantes de cada jugador, las cuales se decrementan si un jugador ha sido impactado por una bala del oponente.



4.2.2. Naves

Las naves están compuestas de imágenes tipo PNG de 100x300 píxeles. En ambas naves predomina el color gris con contorno negro, tienen también colores rojos, naranjas y amarillos para simular fuego saliendo de los motores.

Una de ellas tiene señas de color rojo mientras que la otra las tiene de color amarillo, como se muestra en la siguiente imagen:

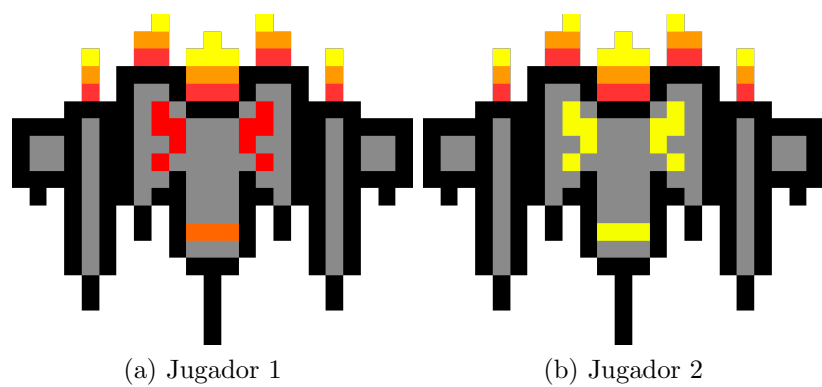


Figura 4.1: Naves

Capítulo 5

Conclusiones

In termino.

Al finalizar este proyecto pudimos lograr comprender de una mejor manera todo el trabajo que conlleva crear un videojuego, así como la lógica detrás de la escritura del código y la manera en la que se debe de observar un juego desde la perspectiva de un desarrollador, es decir, ver el programa tomando en cuenta el paradigma de programación orientada a objetos.

Pudimos observar cómo se van actualizando y modificando los elementos del programa con cada ciclo temporal que pasaba.

También logramos reforzar nuestros conocimientos sobre la programación en Python. Este lenguaje de programación nos ayudó mucho a la realización del programa dada la facilidad que tiene de utilizarse para el paradigma de programación orientada a objetos. Además de ser un lenguaje de programación muy utilizado actualmente, esto significa que al poner en práctica nuestra escritura de código en este lenguaje nos prepara un poco más para nuestro futuro como programadores.

Nos dimos cuenta de que los conocimientos previamente adquiridos a lo largo de nuestra carrera se conjuntaron para ayudarnos a entender y así realizar el proyecto con mucha más facilidad.

Bibliografía

- [1] Sweigart, A. (2015). Inventa tus Propios Juegos de computadora con Python. Libro Version ES. 0.1.
- [2] <https://www.anaconda.com/>
- [3] <https://www.python.org/>
- [4] <https://www.pygame.org/docs/>