

**Estudiante:** Sebastian Bolaños Zamora

**Docente:** Jason Leiton Jimenez

**Curso:** Introducción a la programación

**Carrera:** Ingeniería en computadores

**Universidad:** Instituto Tecnológico  
Costa Rica

---

## Introducción

En esta sección, se presenta una idea general del proyecto, que consiste en la creación de un videojuego inspirado en el clásico Bomberman de NES. Para lograr esto, se emplearán conceptos básicos de Python para el manejo de datos y la interfaz gráfica utilizando Tkinter.

El propósito principal es desarrollar un juego que sea atractivo visualmente y divertido para el usuario. Además, se pretende que este proyecto sirva como una oportunidad de aprendizaje, ayudándome a mejorar mis habilidades en la resolución eficiente de problemas.

Este proyecto es de gran importancia ya que permite demostrar el conocimiento adquirido en el manejo de los aspectos básicos de Python relacionados con el desarrollo de la interfaz gráfica y la gestión de datos, aspectos fundamentales abordados en este trabajo.

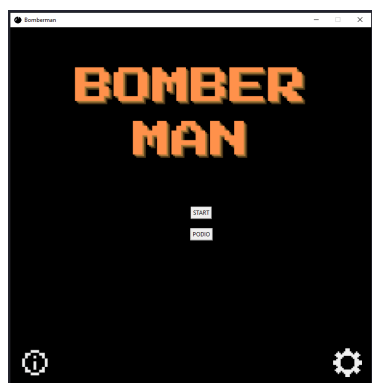
En este documento se abordarán los problemas encontrados durante el proyecto y sus respectivas soluciones. Se presentarán las dificultades encontradas durante su realización, junto con una bitácora detallada de las actividades llevadas a cabo y una estadística de los tiempos empleados en las fases clave del proyecto. Además, se realizará un análisis de resultados. Estos elementos conducirán a las conclusiones del proyecto, las cuales incluirán los logros alcanzados, áreas de mejora y lecciones aprendidas durante el desarrollo del mismo. También se citarán las fuentes utilizadas para ayudar en la elaboración del proyecto.

## Descripción del problema

Este proyecto enfrenta desafíos al crear un videojuego tipo Bomberman usando la biblioteca de Python llamada tkinter. La dificultad radica en que tkinter no está diseñada específicamente para juegos, lo que complica su uso para este propósito. Además, la falta de experiencia en este tipo de proyectos añade otro nivel de dificultad.

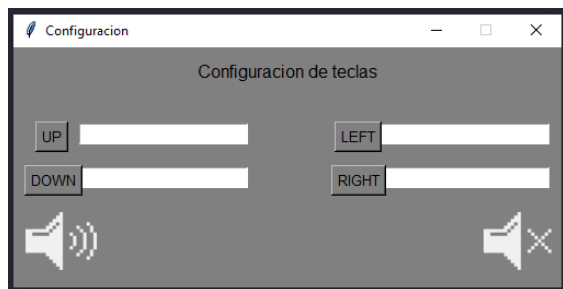
## Análisis de Resultados

Pantalla de inicio:



Esta pantalla es la que aparece al iniciar el juego, en ella hay un canvas el cual contiene todos los elementos (Títulos, botones e imágenes), desde aquí se puede acceder a la información del juego, su configuración, el podio y la pantalla de personalización

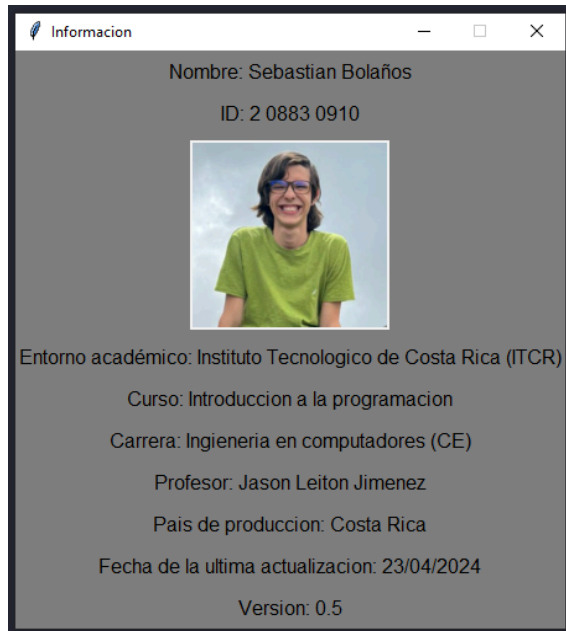
Configuracion:



Esta ventana es accesible desde dentro del juego y desde la pantalla principal.

Su función es cambiar las teclas a gusto propio y manejar la música.

## Informacion:



A esta ventana solo se puede acceder desde la pantalla principal.

La función de esta ventana es mostrar información relevante con respecto al autor, su universidad y el juego.

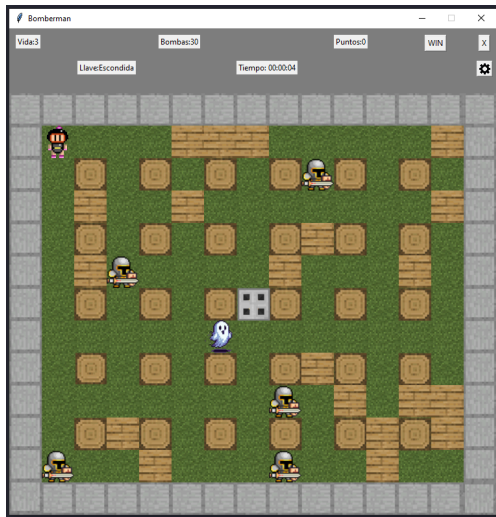
## Personalizar:



Esta ventana aparece al tocar el botón jugar de la pantalla principal.

En esta ventana se ingresa un nombre y se escoge un personaje para que el jugador tenga algo de personalización en su partida.

## Juego:

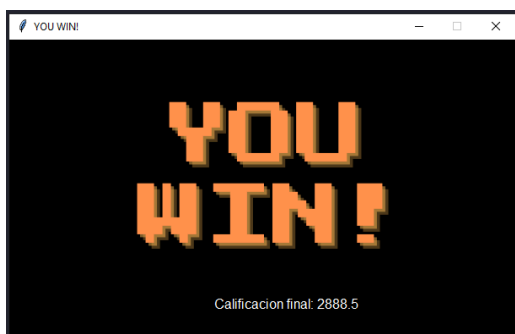


Esta ventana aparece después de la personalización, y aquí es donde sucede toda la magia. Esta ventana está compuesta por 2 canvas, uno que contiene todas las estadísticas del juego y otro en el que se está realizando las acciones del mismo.

La funcionalidad de la parte de las estadísticas es mantener al jugador informado del juego, además se brindan las opciones de configuración y de salir del juego.

La parte del juego como tal tiene varios elementos, tales como: Una imagen de las cosas indestructibles, el personaje, bloques destructibles, caballeros, y un fantasma, cada uno de estos está compuesto por su imagen gráfica y su posición en una lista, la cual se usa para que interactúen entre si y la bomba pueda interactuar con ellos de manera correcta.

## Ganar:



Esta ventana aparece al finalizar los 3 niveles del juego, esta se compone de un canvas con el texto "YOU WIN!" y además se muestra su calificación final

## Perder:



Esta ventana aparece si el jugador llega a perder todas las vidas, aquí solo hay un canvas con el texto “GAME OVER”

## Dificultades Encontradas

### Enemigos:

Mi idea con los enemigos era hacer un tipo de enemigo estacionario y otro que se moviera aleatoriamente, que al final fue lo que logré, sin embargo tuve muchas dificultades con el enemigo aleatorio debido a que no interactuaba bien con la bomba, este problema se debía al manejo de la lista en la que se guardaban los valores del enemigo, al final lo logre solucionar mediante el largo de la lista y no sus elemento exactos ya que por alguna razón no estaban interactuando como yo quería.

### Configuración de controles:

Aquí lo que me estaba causando problemas fueron las mayúsculas y encontrar una manera eficiente de hacerlo, al final logre reducir el bind de las teclas a uno solo e hice una funcion que me ayudara a manejar la configuración de las teclas de manera que hacia la entrada como si fuera una minúscula y después ya se ejecutaba la función de moverse.

---

## Pasar de nivel:

Mi idea con el juego es que a la hora de estar jugando solo hubiera una ventana abierta para mantener la inmersión en el juego, sin embargo esta idea se me cayó abajo a la hora de pasar de nivel, esto debido a que no logre encontrar una manera eficiente de hacer el cambio de nivel en la misma ventana, por lo que tuve que hacer fue una nueva ventana por cada vez que se esté jugando.

## Bomba:

Mi problema con la bomba es que era demasiado destructiva y no interactúa bien con las cosas, cuando digo que era demasiado destructiva es que destruía la llave, al personaje e incluso el fondo del juego, así que para evitar esto tuve que crear una función por separar la cual evalúa el bloque al cual va a destruir y a partir de eso decide si lo quita o no, de esta manera ya no me quitaba las cosas que no quería, sin embargo todavía tenía el problema que no interactúa bien con las listas de enemigos y de objetos destructibles, al final logre implementar un if en la función que evalúa si destruir o no el objeto el cual me ayudaba a quitar los elementos destruidos de las listas

## Configuración del sonido:

Como es una misma ventana para configurar el sonido en 2 ventanas diferentes con músicas diferentes tenía el problema de que en ciertos casos las canciones se mezclaban, entonces tuve que implementar variables para cada ventana y así reproducir la canción correcta

Y así muchas otras las cuales son de menor importancia pero que siempre pesan.

## Bitácora de Actividades

| Dia(s)                       | Actividad                               | Descripción  | Duración |
|------------------------------|---|--|----------|
| Mar 24, 2024<br>Mar 25, 2024 | Creación de sprites                     | Estuve trabajando en el diseño y creación de los sprites para el proyecto (Me hicieron falta los enemigos) | 5h aprox |
| Mar 27, 2024                 | Investigación previa                    | Estuve investigando métodos para realizar el trabajo de manera eficiente                                   | 4h aprox |
| Mar 27, 2024<br>Mar 29, 2024 | Pantalla de inicio                      | Empecé a trabajar la la pantalla de inicio del juego con todos los requisitos                              | 5h aprox |
| 29 mar 2024                  | Pantalla de personalización             | Hice la entrada para el nombre y los botones para elegir personaje   | 2h aprox |
| 8 abr 2024                   | Stats del juego                         | Hice los stats visibles mientras se esta jugando   | 2h aprox |
| 9 abr 2024                   | Pantalla de config                      | Le arregle el audio y añadi el layout para cambiar los controles   | 2h aprox |
| 23 abr 2024                  | Crea las bases del juego                | Definir tamaños y posiciones en el mapa, así como sus bordes y objetos indestructibles                     | 5h aprox |
| 27 abr 2024                  | Objetos destructibles, enemigos y llave | Generación de los objetos destructibles, la llave y los enemigos   | 4h aprox |



Proyecto I – Vintage Bomberman Game

Instituto Tecnológico Costa Rica  
Escuela de Ingeniería en Computadores  
Introducción a la programación (CE1101)

|             |   |   |          |
|-------------|---|---|----------|
| 21 abr 2024 | Bomba y sus interacciones                             | Cree todo lo que tiene que ver con la bomba                       | 5h aprox |
| 1 may 2024  | Progresión de niveles, estados de win o loose y podio | Hice todo lo que sucede después del juego y arregle algunas cosas | 4h aprox |

Estadística de tiempos:

| Actividad   | Duracion |
|---|----------|
| Creación de sprites                                   | 5h aprox |
| Investigación previa                                  | 4h aprox |
| Pantalla de inicio                                    | 5h aprox |
| Pantalla de personalización                           | 2h aprox |
| Stats del juego                                       | 2h aprox |
| Pantalla de config                                    | 2h aprox |
| Crea las bases del juego                              | 5h aprox |
| Objetos destructibles, enemigos y llave               | 4h aprox |
| Bomba y sus interacciones                             | 5h aprox |
| Progresión de niveles, estados de win o loose y podio | 4h aprox |

---

## Conclusion

### Logros Alcanzados:

- 1. Desarrollo de un videojuego:** Se logró crear un videojuego completo inspirado en Bomberman, que incluye múltiples características como pantalla de inicio, personalización de personajes, niveles de juego, pantalla de victoria y derrota, entre otros.
- 2. Manejo de Tkinter para interfaz gráfica:** A pesar de las limitaciones de Tkinter para juegos, se pudo utilizar de manera efectiva para crear una interfaz gráfica atractiva y funcional.
- 3. Resolución de Problemas de Programación:** Se enfrentaron varios desafíos durante el desarrollo del juego, como la interacción de la bomba con otros elementos del juego. Estos desafíos se abordaron con éxito, lo que demuestra habilidades sólidas de resolución de problemas.

### Lecciones Aprendidas:

- 1. Creatividad en la Programación:** Se aprendió a pensar de manera creativa para encontrar soluciones a problemas técnicos.
- 2. Gestión de Proyectos:** Se adquirieron habilidades en la gestión de proyectos de desarrollo de software, incluyendo la planificación, ejecución y seguimiento de tareas, así como la gestión de tiempo y recursos.
- 3. Optimización de Código:** Se comprendió la importancia de escribir código eficiente y optimizado.

### Áreas de Mejora:

- 1. Optimización de Rendimiento:** Se identificó la necesidad de optimizar el rendimiento del juego, especialmente en términos de velocidad y fluidez de la interfaz gráfica.
- 2. Refinamiento de Funcionalidades:** Aunque el juego es funcional, existen áreas que podrían mejorarse, como la inteligencia artificial de los enemigos y la diversificación de los niveles para aumentar la rejugabilidad.

---

## Referencias:

*Documentation.* (s. f.). Python. <https://docs.python.org/>

*Introduction to Python Tkinter.* (2024, 21 abril). [Vídeo]. Youtube.

[https://youtube.com/playlist?list=PLFPWzE\\_xol9kdu\\_Zhtc3zWkGjJPfkKiLS&si=gL5d8JQlcJp6A\\_Ur](https://youtube.com/playlist?list=PLFPWzE_xol9kdu_Zhtc3zWkGjJPfkKiLS&si=gL5d8JQlcJp6A_Ur)

## Video demostrativo:

<https://youtu.be/OQHbVdsbRIE?si=mlSMNyrA3RGCSGnC>