Instituto Tecnológico de Costa Rica

Proyecto I

Juan José Rodríguez Chaves

Gabriel Brenes Casasola

Escuela Ingeniería en Computadores

CE-1104 Fundamentos de Sistemas Computacionales

“Jonathaninho Soccer 64”

Luis Alonso Barboza Artavia

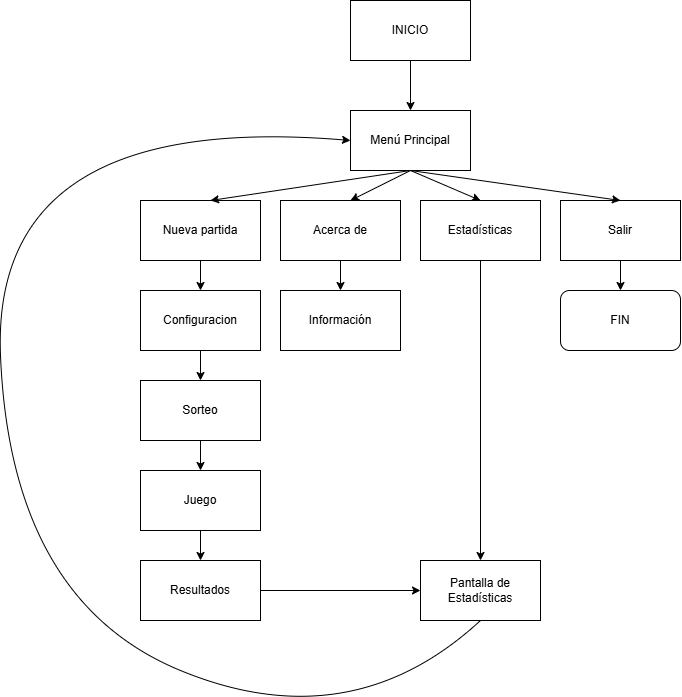
II Semestre de 2025

Introducción:

La implementación de cómo los videojuegos han pasado de un formato más retro a una funcionalidad más compleja de control/software ha sido de hecho bastante reciente. Alrededor de los años 50 se empezó a ir originando lo que vienen siendo los primeros videojuegos controlados a partir de controles, como Nimrod (1951), Oxo (1952) o Spacewar (1962). Actualmente, en la implementación de el videojuego contemplado en el proyecto actual se puede observar la tecnología utilizada de manera básica en gran parte de los videojuegos a un nivel más básico en ingeniería. Implementamos dispositivos IoT como botones, leds, etc. Que se comunican con un programa a partir de un sistema de empotrado, en este caso, una Raspberry Pi Pico W.

Análisis:

Diagrama de flujo del programa Jonathaninho Soccer

Para la realización del proyecto se debe primero explicar y analizar la funcionalidad de este. A través del diagrama de flujo expuesto en la ilustración 1; el juego comienza en la pantalla de inicio al iniciar el programa y de esta manera se observan distintas opciones para el siguiente flujo. En este caso se pueden seleccionar entre una nueva partida, la pantalla de información y de estadísticas, además del botón para salir del juego.

Ilustración

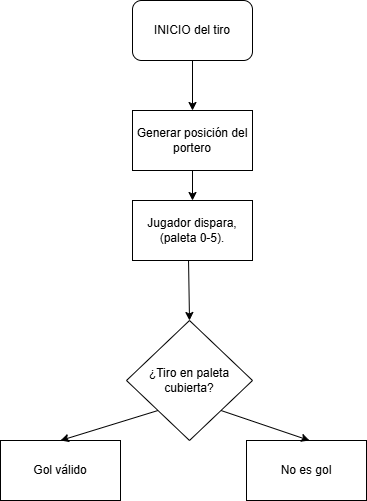
Dentro de la pantalla de configuración se encontraría la elección de los equipos, jugadores, porteros y la que se distingue de ser para el modo manual o automático. Luego de que el usuario haya terminado la decisión se lleva a cabo un sorteo, ilustrado a partir de una animación de moneda, de esta manera, se evidencia quien será el equipo local y el equipo visitante a partir de una elección aleatoria de 50/50 y luego de esto, empieza el juego. El flujo de este se puede observar en la ilustración 2 y 3. Primeramente el portero elige entre tres tipos de índices de entre 2 y 3 paletas cubiertas, ya sea juntas o alternadas y se genera la posición. Luego de esto, se hace la decisión de si la posición a la que se tiró el balón fue cubierta o no por el portero. De esta manera se decide si fue un gol o no.

Diagrama de comportamiento del portero

El juego siempre empezará luego del pito del árbitro, el cual se reproduce después de un cooldown y dependiendo del modo, (automático o manual). Se decidirá lo que pasará luego de un tiro.

Ilustración

El software se compone de un main.py que dirige una clase de configuración, audio y hardware. Además, dirige las distintas pantallas organizadas en su propia carpeta, una para la información, otra para la pantalla principal del juego, la pantalla de elección de jugadores, pantalla de sorteo, pantalla de estadísticas y pantalla de juego y una clase auxiliar que maneja las estadísticas la cual se ubica con el archivo que dirige esta desde la pantalla organizado de la siguiente manera:

Sistema principal:

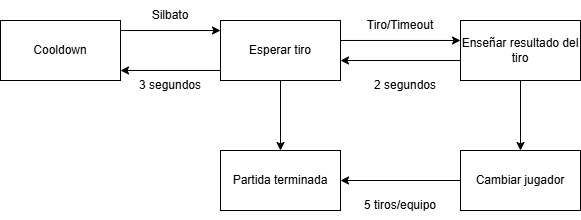
* main.py
* config.py
* hardware\_manager.py
* screens
  + about\_screen.py
  + coin\_toss\_screen.py
  + config\_screen
  + game\_screen
  + instructions\_screen.py
  + main\_menu.py
  + stats\_manager.py
  + stats\_screen.py
* images
* sounds
* docs

Diagrama de flujo general del juego

Ilustración

Ahora bien, el hardware funciona de maneras bastantes diferentes con un código hecho en este caso con MicroPython. A través de los siguientes diagramas se puede observar de forma general el flujo y funcionalidad de los componentes físicos y como se comunican con el software.A diagram of a flowchart

AI-generated content may be incorrect.

Ilustración 3

Bitácora:

**4 de octubre de 2025:**

Se adelantó el proyecto desde el principio iniciando con una pantalla de menú principal y una pantalla de “acerca de” que mencionará la información de los creadores junto con sus fotos. Luego se implementó la pantalla de instrucciones, una pantalla simple que explica la forma de jugar al videojuego con el teclado. Además, se implementó un archivo especial para toda configuración. Luego de esto, se implementó la música de fondo, al principio del proyecto solo se reproducía en la pantalla de inicio

**9 de octubre de 2025:**

La implementación inicial fue de la pantalla de configuración de los jugadores de manera inicial, sin fotos, sin nombres en específico, solo a partir de un Shooter1, 2 y 3. Además se trabajó en ciertos bugs en la pantalla de instrucciones que no dibujaban la imagen de fondo. Posterior a esto se empezó a agregar las imágenes responsables de los equipos

**14 de octubre de 2025:**

La música de fondo se reemplazó a la actual y no hubo grandes cambios más que en el menú principal y la pantalla de los autores.

**15 de octubre de 2025:**

Se empezó la implementación de la pantalla del sorteo de moneda para elegir qué equipo será el local y qué equipo será el visitante. Se ajustaron ciertos cambios en varias otras pantallas, la de configuración de los jugadores y la de menú principal.

**16 de octubre de 2025:**

La pantalla de instrucciones fue completamente corregida, ahora sí se dibuja la imagen de fondo.

**17 de octubre de 2025:**

Las imágenes pertenecientes a cada jugador de cada equipo empezaron a ser implementadas, empezando por los tiradores del Real Madrid, Chelsea y Barcelona. Además, se ajustaron levemente casi todas las pantallas para poder implementar de manera efectiva las imágenes. Adicional a esto, se comenzó a implementar de manera temprana la pantalla del juego, en la cual se dibujaría la cancha del juego.

**19 de octubre de 2025:**

Se agregó Gitignore y se agregan los sonidos pertenecientes a los goles, abucheos y nueva música de fondo para el sorteo y el juego.

**20 de octubre de 2025:**

Adición de las fotos de los porteros, cambio en la posición de los botones en la pantalla de configuración, etc. Se implementó además un cooldown, requerimiento del juego según el documento antes de cada turno, hubo arreglo de bugs, y el sonido del silbato se arregla.

**21 de octubre de 2025:**

Se implementa de manera temprana el modo manual y automático y su funcionalidad en la partida del juego, sin embargo, no logró ser terminado completamente.

**22 de octubre de 2025:**

La funcionalidad del proyecto referente al programa/software está casi completamente integrada. Pues se completó la implementación de las estadísticas, pantalla de juego, menú principal, manejador de estadísticas, etc.

**24 de octubre de 2025:**

No hubo grandes cambios, se empezó a implementar ciertos archivos .md referentes a la documentación.

**26 de octubre de 2025:**

El proyecto comienza a tomar forma completamente, la funcionalidad de los botones y del potenciómetro está completa para la rotación de opciones en la pantalla de configuraciones, botón 2 para iniciar el juego y botón 3 para volver al menú principal. El código de MicroPython comienza e implementa de manera satisfactoria casi todos los detalles.

**27 y 28 de octubre de 2025:**

La documentación es adelantada a su forma casi final dejando ciertos detalles incompletos que son pulidos. Desafortunadamente la Raspberry tuvo un cortocircuito y se tuvo que posponer ciertos detalles del 27 hasta el 28 de octubre. Para este momento, se conectó completamente los LEDs, potenciómetro, botones y batería. El código de la Raspberry Pi Pico W es finalizado junto con el del software y se implementan detalles como el archivo README.md. Desde el 27 hasta el 28 se intenta crear la estructura de la maqueta final y se completa.

Uso de repositorio:

<https://github.com/juanrckx/JonathaninhoSoccer64v.01/tree/master>

(Se le detalla al profesor que el compañero Gabriel no posee cuenta de Github, por lo tanto, todo se llevó a cabo desde la cuenta de Juan José)

Conclusiones:

Se logró concretar un sistema completo que conecta circuitos simples a través de un microprocesador y arquitectura modular que se conecta con Wifi a un software completamente realizado por los estudiantes con una interfaz de usuario pulida y profesional. Además, la integración software/hardware fue completamente exitosa, gracias a distintas librerías pertenecientes a los lenguajes de Python y Micropython.

Los detalles han sido de suma importancia, como la música, efectos de sonido, animaciones, gestión de estadísticas y la perseverancia de estas, etc. El proyecto entregó una valiosa aplicación práctica de conceptos de programación vitales y levemente complejos, utilizando por ejemplo POO.

Recomendaciones:

El proyecto es satisfactorio, sin embargo, podría recomendarse añadir ciertas mejoras, como un sistema de rankings online del videojuego, diferentes niveles de dificultad, modo torneo y más equipos y jugadores personalizables. Podría mejorarse además el rendimiento de algunas animaciones.

Referencias:

CodeLab. (Julio 25 de 2024). *Raspberry Pi Pico W Pinout Explained: A Detailed Guide with Practical Examples.* <https://rajivcodelab.com/raspberry-pi-pico-w-pinout-explained>

Telefonica.(18 de noviembre de 2024). *Historia de los videojuegos.* <https://www.telefonica.com/es/sala-comunicacion/blog/videojuegos-historia/>