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

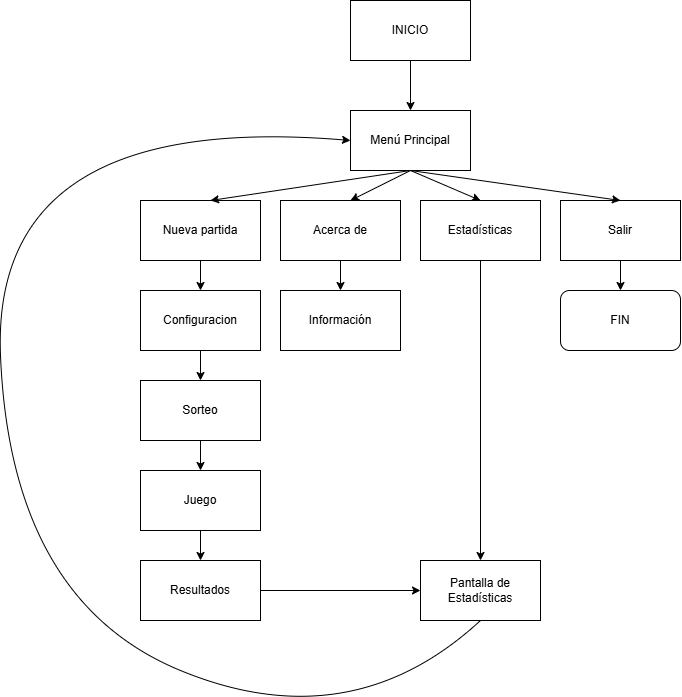
Para la realización del proyecto se debe primero explicar y analizar la funcionalidad de este. A través del diagrama de flujo expuesto en la ilustración 1; el juego comienza en la pantalla de inicio al iniciar el programa y de esta manera se observan distintas opciones para el siguiente flujo. En este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

Ilustración 1

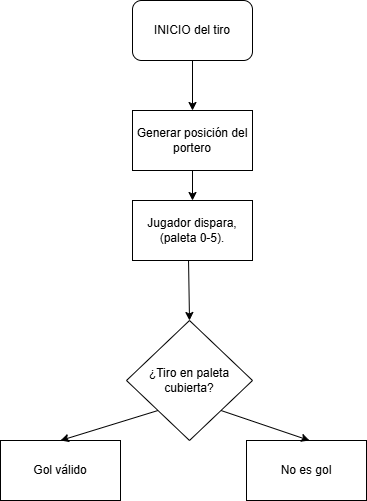
Dentro de la pantalla de configuración se encontraría la elección de los equipos, jugadores, porteros y la que se distingue de ser para el modo manual o automático. Luego de que el usuario haya terminado la decisión se lleva a cabo un sorteo, ilustrado a partir de una animación de moneda, de esta manera, 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 1

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 2

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:

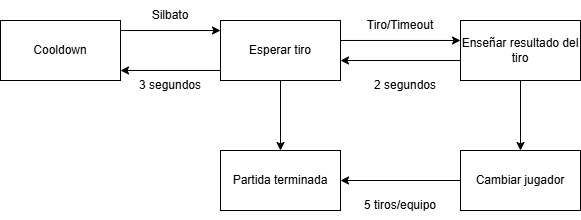
* main.py
* config.py
* hardware\_manager.py
* screens
  + about\_screen.py
  + coin\_toss\_screen.py
  + config\_screen
  + game\_screen
  + instructions\_screen.py
  + main\_menu.py
  + stats\_manager.py
  + stats\_screen.py
* images
* sounds
* docs

Diagrama de flujo general del juego

Ilustración 3

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.

Bitácora:

Uso de repositorio:

Conclusiones:

Recomendaciones:

Referencias:

<https://rajivcodelab.com/raspberry-pi-pico-w-pinout-explained>

https://www.telefonica.com/es/sala-comunicacion/blog/videojuegos-historia/