

**Alejandro Henao Cordero**

**Christian Navarro Ellerbrock**

**Jorge Gutiérrez Vindas**

## **Proyecto 1. Algoritmos y Estructuras de Datos I**

### **Documentación de diseño**

#### **a. Listado de requerimientos del sistema:**

- El sistema requiere de un servidor que maneje el funcionamiento de todo el juego menos las decisiones que tienen que ser tomadas por el mismo jugador dentro del juego.
- El sistema requiere de una interfaz gráfica donde el jugador pueda dibujar y ver las líneas de la malla de puntos, esta tiene que estar incorporada con el cliente.
- El sistema requiere de un control el cual pueda permitir al jugador seleccionar y dibujar las líneas del juego.
- El sistema requiere de un servidor que contenga a todos los clientes que se quieran añadir y ponerlos a jugar.
- El sistema requiere de diferentes sockets para el envío y recibo efectivo de información entre el servidor y los clientes.
- El sistema requiere identificar la puntuación que se tiene cada jugador para poder decidir un ganador al final.

#### **b. Elaboración de opciones de solución al problema:**

En el sistema, se tiene el problema de que la elaboración del control se puede realizar de múltiples maneras. El control es una parte importante en el desarrollo del proyecto ya que, gracias a este componente de software, se pueden dibujar las líneas necesarias para crear los cuadrados y ganar el juego.

- Solución 1: Se plantea como posible solución la creación de un control el cual es uno ya realizado por otra compañía para jugar videojuegos en una consola y adaptarlo para que este se pueda programar y utilizar mediante Raspberry Pi. Para la conexión del hardware y software, se utilizaría un adaptador USB inalámbrico el cual pueda conectar con el dispositivo donde se tendrá el código para el funcionamiento del control. La parte de software se implementaría directamente en Java mediante la utilización de una librería como Pi4J de Java. Ya en funcionamiento, el control se movería con un joystick para poder ir de punto en punto en la malla y seleccionar los puntos que el jugador quiere unir con uno de los botones para así crear las líneas necesarias para ganar el juego. Finalmente, las acciones realizadas en el control son enviadas a los clientes del servidor para que pueda conectarse el control con el juego en sí.

- Solución 2: Se plantea como posible solución la implementación de un control adquirido específicamente para la creación de este proyecto. Este control funcionaría utilizando Arduino y se implementaría como si se estuviese emulando varios componentes de la computadora en la que se aplica el juego. Esto se implementa usando el lenguaje de programación Python. Ya en el funcionamiento, el control tiene un Joystick el cual emula el cursor de la computadora para ir cambiando de punto en punto y elegir los puntos que se van a unir para formar una línea. Además, se tiene un botón para poder seleccionar los puntos específicos que el jugador desea unir. Finalmente, las acciones realizadas en el control son enviadas a los clientes del servidor para que pueda conectarse el control con el juego en sí.

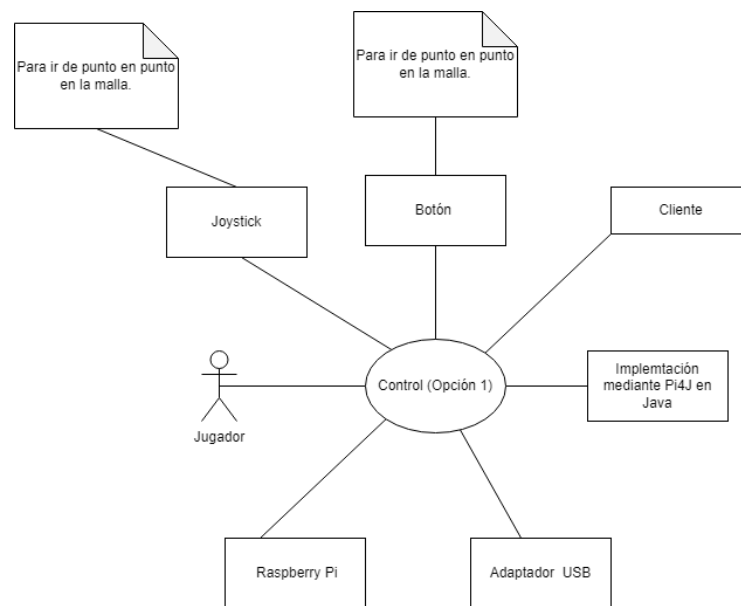


Diagrama 1: UML de contexto de la opción 1

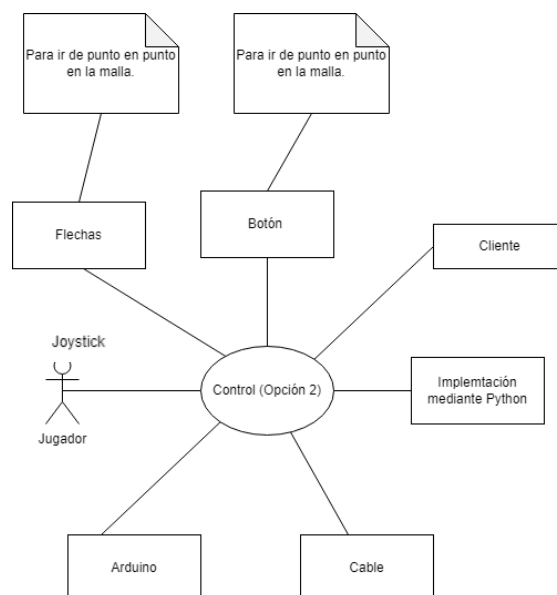


Diagrama 2: UML de contexto de la opción 2

**c. Valoración de opciones de solución:**

Valoración de la solución 1:

- No se necesita adquirir demasiadas nuevas piezas de hardware en el control en sí ya que se reutiliza un control.
- El sistema de Raspberry Pi es compatible con Java usando ciertas librerías como Pi4J.
- El hardware del Raspberry Pi no es muy grande, por lo que es fácil de adaptar al resto de hardware.
- Trae incorporado y funciona mediante un sistema operativo.
- No se tiene una claridad absoluta sobre el adaptador que conectaría el software con el hardware.
- No se tiene familiaridad sobre cómo implementar Raspberry Pi en este tipo de proyectos.

Valoración de la solución 2:

- Al ser el hardware del control específico para el proyecto, no se necesitan hacer adaptaciones grandes para poder implementar las funcionalidades a Arduino, se simplifica la construcción del hardware.
- El sistema de Arduino es compatible al software en Java con librerías como RXTX.
- La IDE para usar Arduino es compatible con todas las plataformas populares: Windows, Linux y MacOS.
- El sistema de Arduino es relativamente fácil de utilizar.
- Puede llegar a ser un poco cara la implementación en Arduino.
- Se tiene mayor familiaridad y práctica en la utilización de esta herramienta.

**d. Selección de la propuesta final:**

Finalmente, se decidió optar por la solución 2 de las propuestas anteriormente. Primero, esta opción a pesar de que se no se reutilizarían los materiales para hacer el control, este sería más fácil de realizar debido a que se puede configurar de manera más clara. Además, se escogió utilizar Arduino para la implementación ya que se tiene una mayor familiaridad con este sistema de adaptación del hardware al software.

**e. Diseño de la alternativa seleccionada:**

Protocolo

El protocolo utilizado por el código que genera el control es el que se le conoce como el protocolo de red, el cual consiste en el envío de información mediante una red de información, en este caso, en enviar datos entre clientes y un servidor en Java

Diagramas del diseño final:

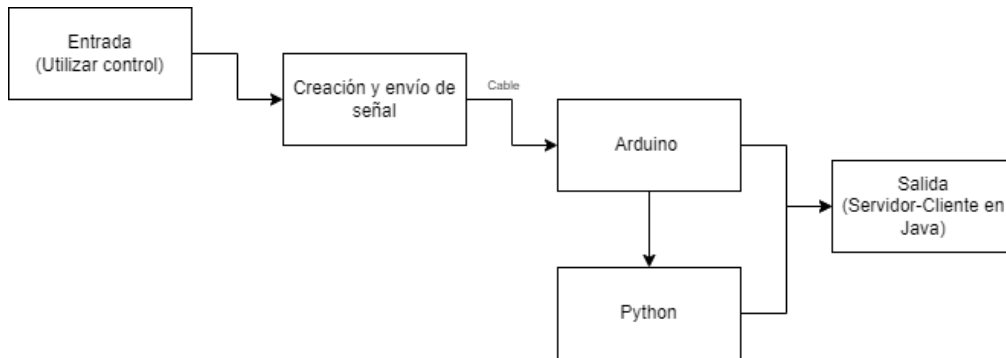


Diagrama 3: Diagrama de bloques del sistema de la alternativa seleccionada

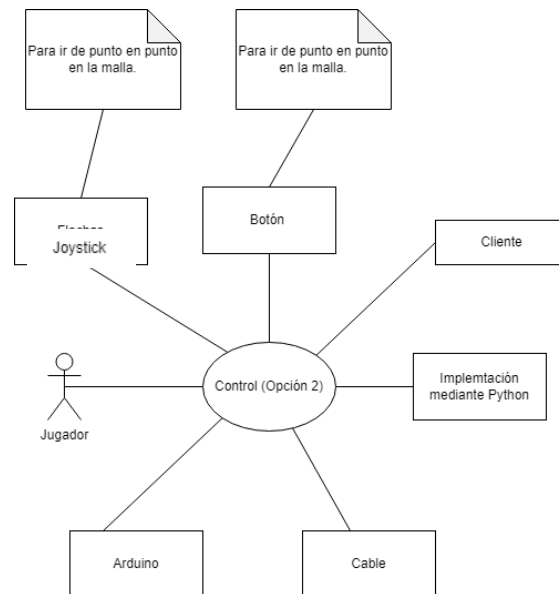


Diagrama 4: Diagrama de contexto UML de la opción seleccionada

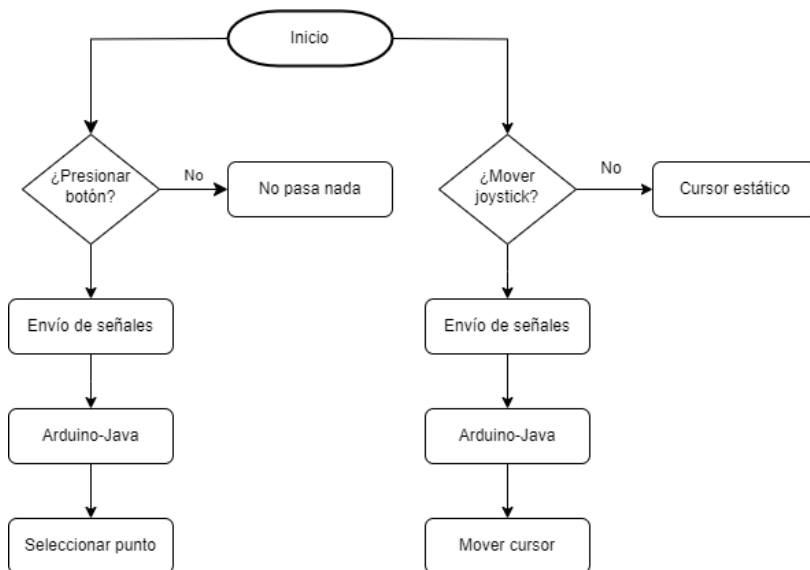


Diagrama 5: Diagrama de flujo UML de la opción seleccionada

#### **f. Validación del diseño**

El diseño final del control tiene un funcionamiento fluido, al mover el joystick no se traba y recorre toda la pantalla de buena manera. Al presionar los botones se detecta de manera correcta los puntos y al seleccionar otro se crea la línea que el jugador quiere insertar. Al final, para incrementar la comodidad del control, se implementó un case hecho de su mayoría por materiales reciclables por lo que el control al final también demuestra un buen cuidado al medio ambiente.