**UNIVERSIDAD DE TARAPACÁ**

**FACULTAD DE INGENIERÍA**

**DEPARTAMENTO DE INGENIERÍA CIVIL EN COMPUTACIÓN E INFORMÁTICA**

Imagen que contiene Forma

Descripción generada automáticamente

**Plan de Proyecto**

**“GTP-1”**

**Alumno(os): Pattricio Medina**

**Jonathan Orellana**

**Alex Muñoz**

**Benjamin Flores**

**Asignatura: Proyecto I**

**Profesor: Humberto Urrutia**

**MES – AÑO**

**Historial de Cambios**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor(es)** |
| 11/11/2024 | 1.0 | Avance del proyecto | **Benjamín Flores** |

Tabla de Contenidos

[1.](#_30j0zll) Panorama General 4

[1.1.](#_1fob9te) Introducción 4

[1.2.](#_3znysh7) Objetivos 4

[1.2.1.](#_2et92p0) Objetivo General 4

[1.2.2.](#_tyjcwt) Objetivo Específico 4

[1.3.](#_3dy6vkm) Restricciones 4

[1.4.](#_1t3h5sf) Entregables 4

[2.](#_4d34og8) Organización del Personal 4

[2.1.](#_2s8eyo1) Descripción de los Roles 4

[2.2.](#_17dp8vu) Personal que cumplirá los Roles 4

[2.3.](#_3rdcrjn) Mecanismos de Comunicación 4

[3.](#_26in1rg) Planificación del Proyecto 4

[3.1.](#_lnxbz9) Actividades 4

[3.2.](#_35nkun2) Asignación de Tiempo 4

[3.3.](#_1ksv4uv) Gestión de Riesgos 4

[4.](#_44sinio) Planificación de los Recursos 4

[4.1.](#_2jxsxqh) Hardware 4

[4.2.](#_z337ya) Software 4

[4.3.](#_3j2qqm3) Estimación de Costos 4

[5.](#_1y810tw) Análisis y Diseño 4

[5.1.](#_4i7ojhp) Especificación de Requerimiento 4

[5.2.](#_2xcytpi) Arquitectura 4

[5.3.](#_1ci93xb) Interfaz 4

[6.](#_3whwml4) Implementación 4

[6.1.](#_2bn6wsx) Fundamentos de Proyectiles 4

[6.2.](#_qsh70q) Descripción de los programas 4

[6.3.](#_3as4poj) Diagramas 4

[7.](#_1pxezwc) Resultados 4

[7.1.](#_49x2ik5) Estado Actual del Proyecto 4

[7.2.](#_2p2csry) Problemas Encontrados y Solución Propuesta 4

[8.](#_147n2zr) Conclusión 4

[9.](#_3o7alnk) Referencias 4

# **Panorama** **General**

## Introducción

## 

## En el siguiente proyecto se descubrirá los procedimientos realizados para la construcción del **robot Lego Ev3** la cual simularemos una garra de carga. El objetivo con el cual se espera conseguir con este proyecto es que el robot sea capaz de moverse con una interfaz y agarrar una pelota con una garra construida. Todos los algoritmos creados son introducidos en el programa **Visual Studio Code** con el lenguaje de **Python**, mediante un **firmware llamado Ev3Dev** que se instala dentro del robot mencionado anteriormente.

## Objetivos

### Objetivo General

### Desarrollar un plan de construcción de un robot Ev3, con el cual es codificado y encargado por estudiantes del departamento de Computación e informática.

### Objetivo Específico

### Planificar el tipo de robot a crear.

### Construir un robot con el kit EV3 Mindstorms.

### Instalar el firmware dentro del robot.

### Realizar un software con algoritmos realizados con el lenguaje python para el movimiento del robot.

### Implementar una interfaz amigable para un manejo sencillo para el usuario.

## Restricciones

* El lenguaje de programación utilizado dentro del Visual Studio Code será Python.
* Un límite de tiempo para la construcción del robot y su programación.
* Todos los documentos (Bitácoras, Carta Gantt e Informe) relacionados al proyecto deberán ser subidos a la plataforma Redmine.
* El sistema operativo utilizado será Linux.

## Entregables

Los archivos que se entregarán durante este proyecto son los siguientes:

* Bitácoras semanales entregadas todos los domingos.
* Informe de plan de proyecto.
* Presentación de plan de proyecto.
* Carta Gantt.
* Actualización del proyecto en Redmine.

Estos documentos, principalmente las bitácoras e informes, serán subidos a la plataforma [Redmin](http://pomerape.uta.cl/redmine/projects/grupo-3-a-2023)e de la Universidad

# **Organización del Personal**

## Descripción de los Roles

En este punto mostraremos los roles asignados para cada integrante del grupo

* + - **Diseñador:** Encargado principalmente del diseño de: Logotipos, Presentaciones, Manual de usuario, etc....
    - **Documentación:** Es el que se encargar en documentar todo lo que se realiza en el proyecto, los avances, informes y bitácoras y carta Gantt.
    - **Jefe de grupo:** El que representa al equipo de trabajo y mantener la organización e iniciativa del mismo
    - **Programador:** Se encarga de crear e implementar los algoritmos necesarios para el movimiento y funcionamiento del robot.
    - **Ensamblador:** Encargado del montaje y armado de robot al igual que monitora el cumplimiento de las funciones del robot.

## Personal que cumplirá los Roles

Las personas encargadas de cumplir los roles anteriormente mencionados son las siguientes:

* + - Jefe de grupo: Alex Muñoz.
    - Diseñador: Pattricio Medina.
    - Documentación: Benjamin Flores.
    - Ensamblador: Alex Muñoz.
    - Programador: Jonathan Orellana.

## Mecanismos de Comunicación

Los medios telemáticos que se utilizarán para la comunicación son Whatsapp para cosas específicas y Discord, siendo Discord el más importante por su fácil uso de creación de grupo, junto con su chat de texto y voz y la facilidad de poder enviar todo tipo de archivos con la cual se puede crear una gran organización de la información.

# **Planificación del Proyecto**

## Actividades (Carta Gantt)



## Asignación de Tiempo

* Planificación de proyecto: 3-4 Semanas
* Planificación de construcción: 1 Semana
* Planificación de interfaz: 1 Semana

## Gestión de Riesgos

Niveles de impacto:

* 1. Catastrófico
* 2. Critico
* 3. Circunstancial
* 4. Irrelevante

|  |  |  |  |
| --- | --- | --- | --- |
| **Riesgos** | **Probabilidad de ocurrencia** | **Nivel de impacto** | **Posibles soluciones** |
| Falta de pieza en el armado del robot | 20% | 3 | Buscar en bodega la pieza faltante, en caso de no estar usar una pieza similar. |
| Daño en la tarjeta SD | 10% | 1 | Cambiar la tarjeta SD por una nueva e instalar de nuevo el sistema operativo. |
| Error en la codificación | 60% | 2 | Investigar nuevamente en sitios webs oficiales, para buscar el posible error y como solucionarlo |
| Enfermedad de algún integrante del equipo. | 60% | 3 | Reorganizar el equipo de tal forma que se pueda cubrir en su totalidad la labor  asignada a dicho miembro. |
| Fallo en el diseño del robot. | 30% | 2 | Realizar un cambio en  el diseño del robot  para que funcione  correctamente. |
| Uno o más miembros dejan el proyecto. | 10% | 1 | Reorganizar las tareas y los roles de cada integrante. |
| Catástrofes naturales. | 10% | 1 | Dependiendo del daño causado, el equipo debería tratar de reunirse de manera  remota o presencial. |
| Pérdida total de archivos o procesos. | 10% | 1 | Recrear todo lo perdido, basándose en el conocimiento adquirido. |
| Quedarse sin batería del robot. | 20% | 3 | Conectarlo a una fuente eléctrica y que se recargue. |

# **Planificación de los Recursos**

## Hardware

El hardware usado en este proyecto fueron los siguientes:

* + - Tarjeta MicroSD.
    - Robot EV3 Mindstorm.
    - Wi-fi Dongle.
    - Notebook.
    - Adaptador MicroSD.

## Software

El software usado en este proyecto fueron los siguientes:

* + - Visual Studio Code
    - Discord
    - Ubuntu
    - Canva
    - Word
    - EV3 dev (ev3dev.org)
    - WhatsApp
    - Python

## Estimación de Costos

|  |  |  |  |
| --- | --- | --- | --- |
| **Productos** | **Cantidad** | **Precio** | **Categoría** |
| Notebook | 1 unidades | $700.000 | Hardware |
| Kit Lego MINDSTORMS (EV3) | 1 unidad | $1.000.000 | Hardware |
| Micro SD (8 GB) | 1 unidad | $5.000 | Hardware |
| Dongle USB Wifi | 1 unidad | $7.000 | Hardware |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Personal** | **Valor horas trabajadas** | **Horas trabajadas** | **Horas extras trabajadas** | **Horas totales trabajadas** | **Sueldo mensual** |
| **ALEX MUÑOZ** | 34.000 | 28 | 4 | 32 | $ 1.088.000 |
| **BENJAMIN FLORES** | 26.000 | 28 | 4 | 32 | $ 832.000 |
| **PATTRICIO MEDINA** | 32.000 | 28 | 4 | 32 | $ 1.024.000 |
| **JONATHAN ORELLANA** | 32.000 | 28 | 4 | 32 | $ 1.024.000 |
| **COSTO TOTAL**  **(MENSUAL)** | X | X | X | X | **$3.968.000** |

# **Análisis y Diseño**

## Especificación de Requerimiento

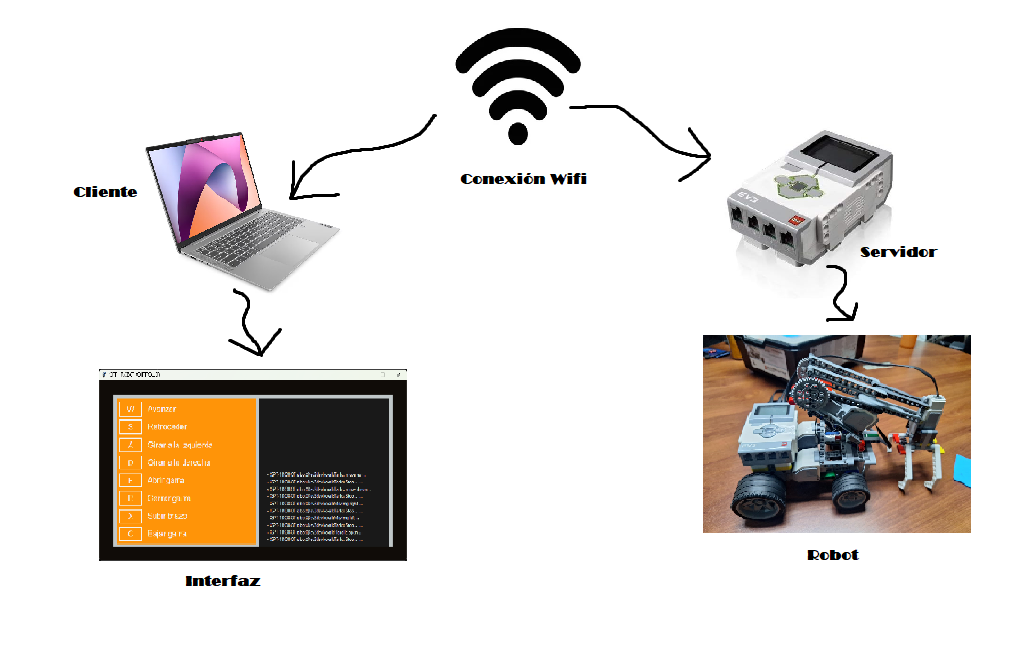
Requerimiento Funcionales:

* El robot tiene la capacidad de ser controlado con una interfaz gráfica de manera remota
* El robot tiene la capacidad de poder moverse
* El robot tiene la capacidad de mover una garra
* El robot tiene la capacidad de poder agarrar una pelota.

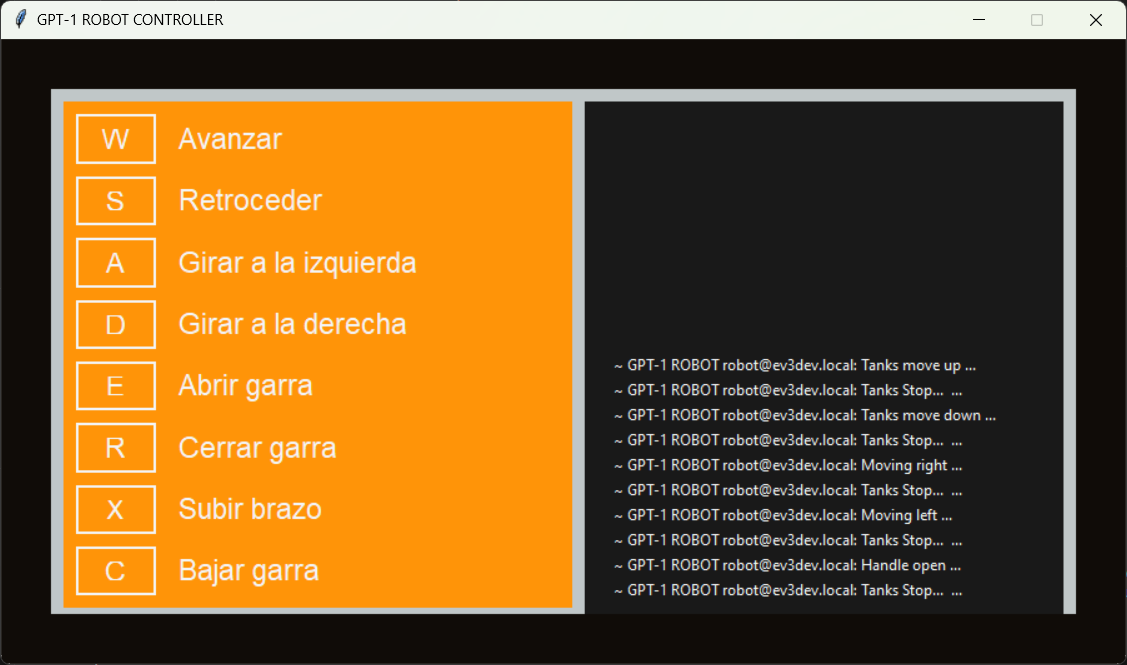
Requerimientos no funcionales:

* Cada movimiento del robot se realizará en un tiempo y velocidad estimados.
* La elaboración del código debe ser con el lenguaje de programación Python
* La interfaz gráfica debe tener todos los movimientos del robot tales como moverse hacia adelante, hacías atrás y hacia los lados, además de mover la garra.
* El robot debe ser construido con las piezas de lego kit Mindstorm EV3 Education.

## Arquitectura



## Interfaz



# **Implementación**

## Fundamentos de Carga

*Cinemática del movimiento*

Desplazamiento lineal

* Posición: Ubicación de la carga o robot
* Velocidad: Rapidez del robot o de la carga
* Aceleración: Cambio de la velocidad del movimiento

Formulas:

* d: Desplazamiento lineal
* v: Velocidad inicial
* t: Tiempo
* a: Aceleración lineal

*Dinámica y fuerzas involucradas*

Fuerzas principales:

* Fuerza de tracción: Generada por motores que impulsan el movimiento
* Fricción: Entre las ruedas y suelo que afecta el desplazamiento
* Fuerzas inerciales: Cuando el robot cambie de velocidad estas fuerzas tienden a resistir el movimiento.

Formulas:

* m: Masa de la carga
* a: Aceleración

*Centro de gravedad*

La posición del centro de gravedad del robot y la carga es crucial ya que evita que el robot llegue a volcarse.

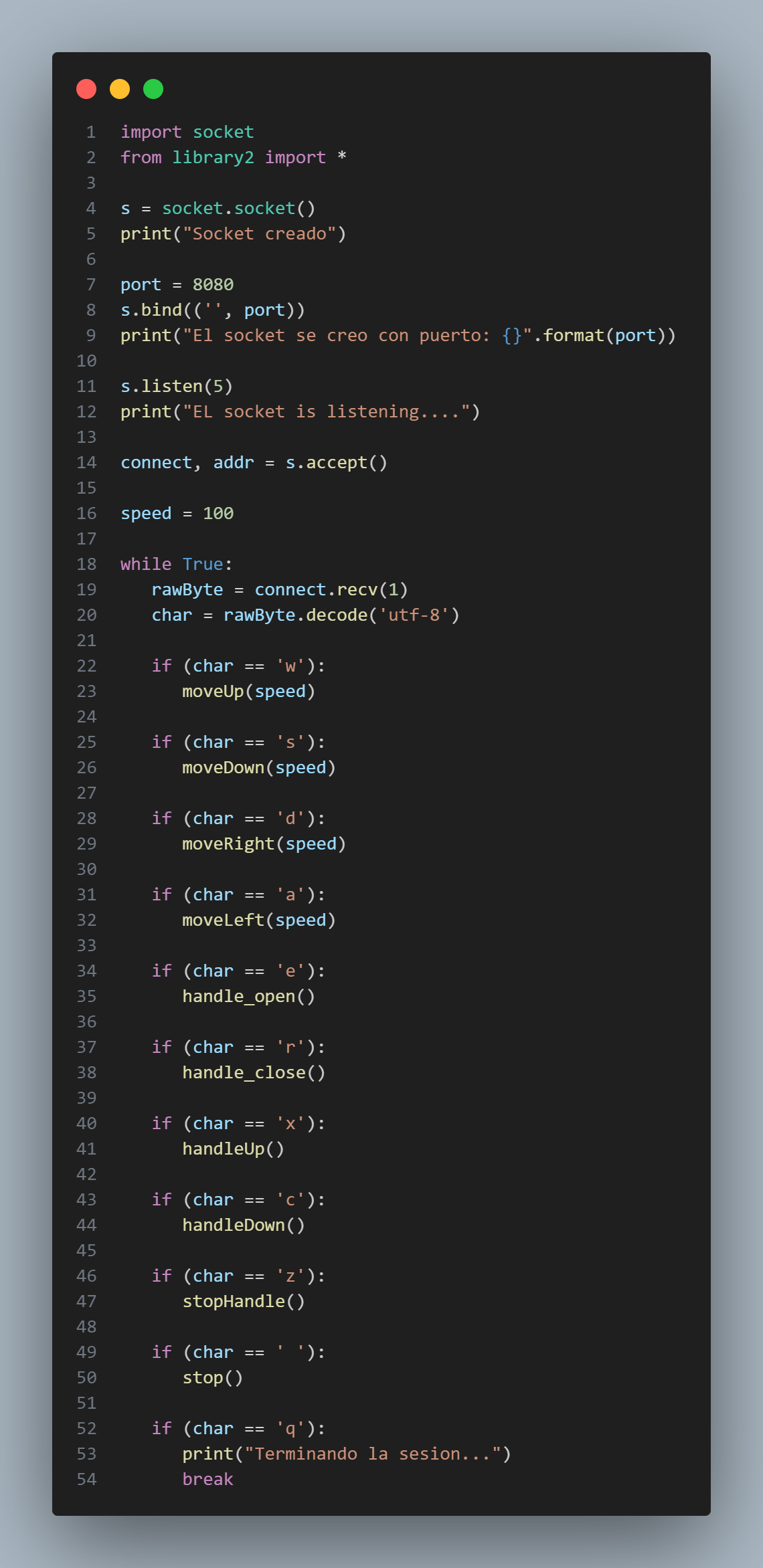
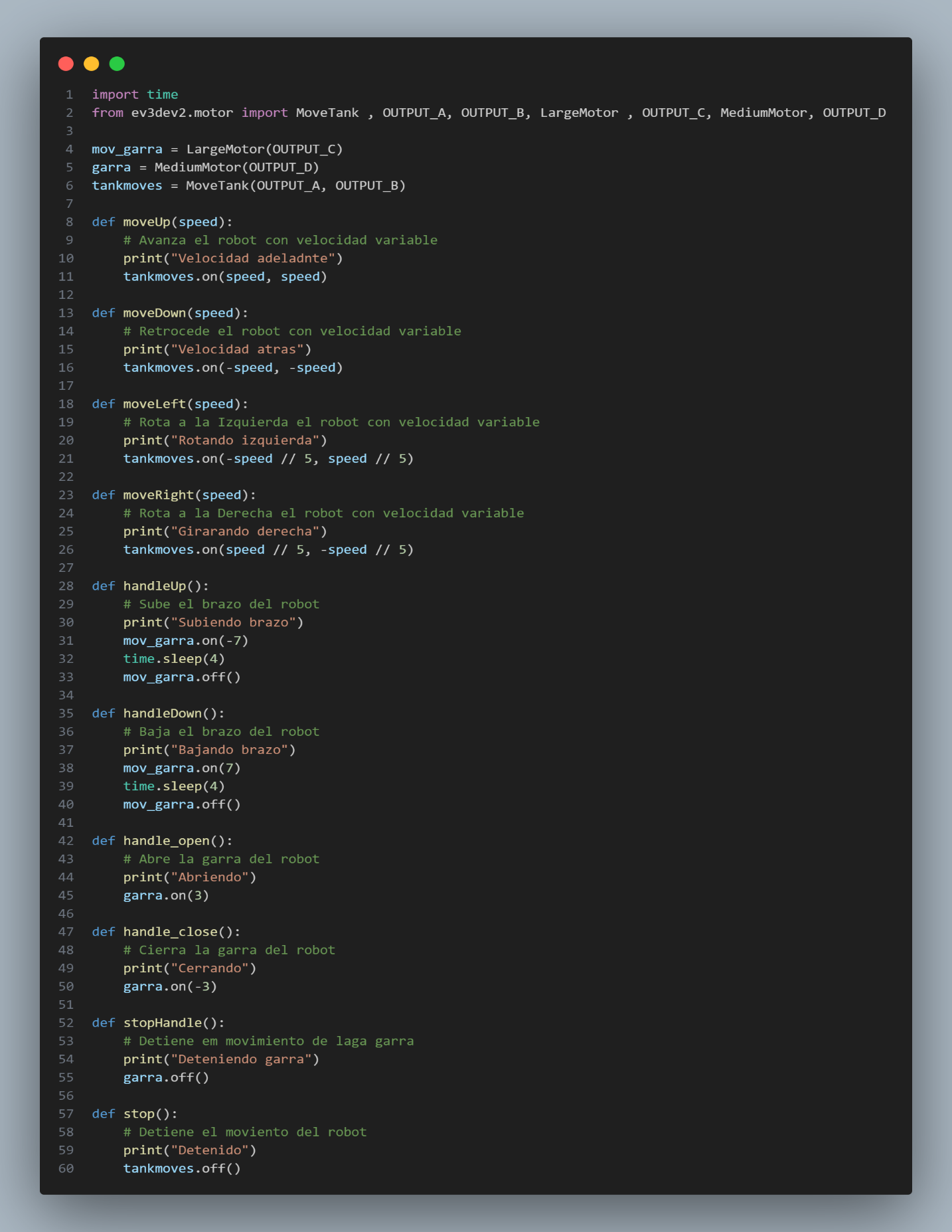
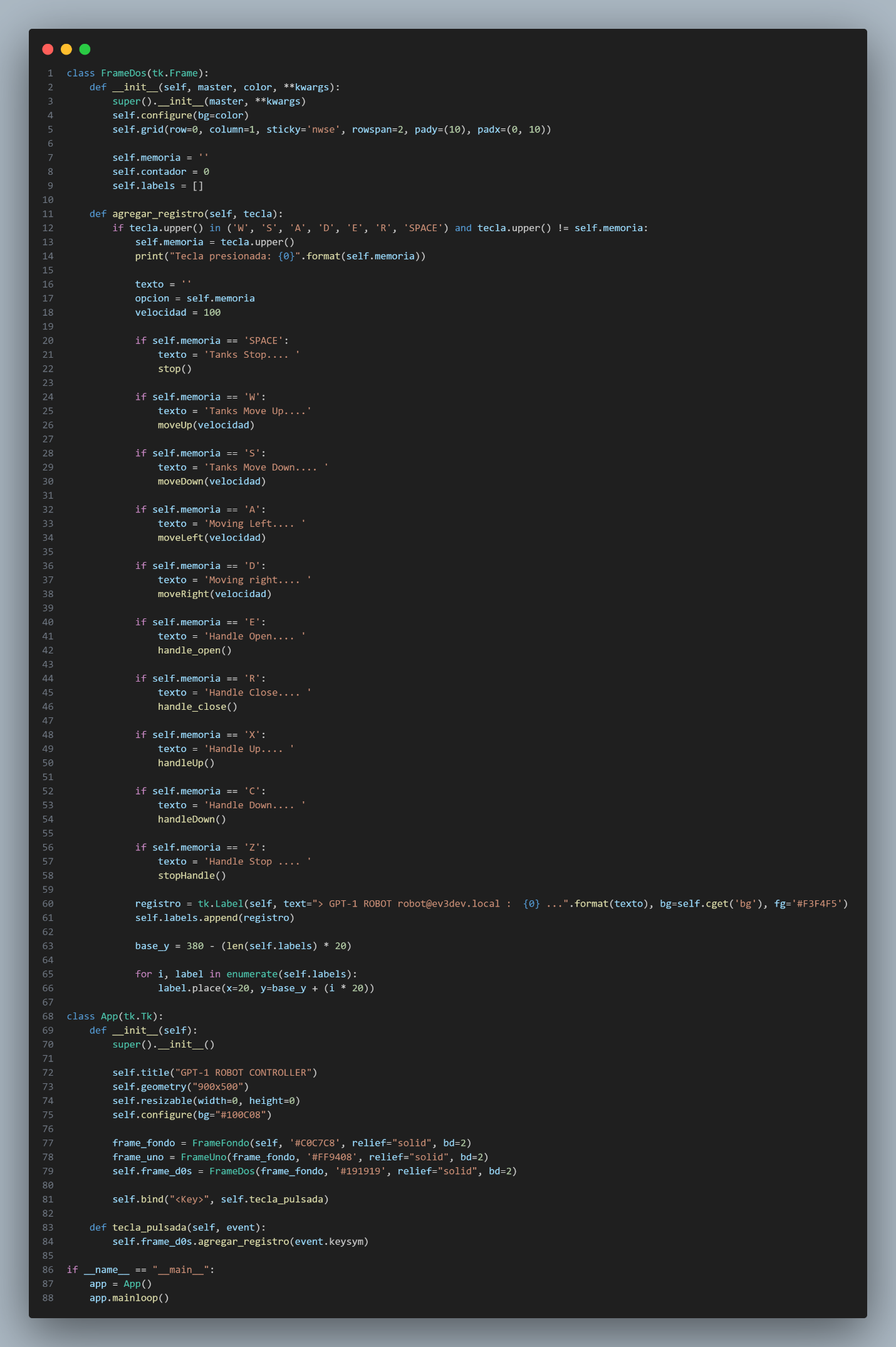
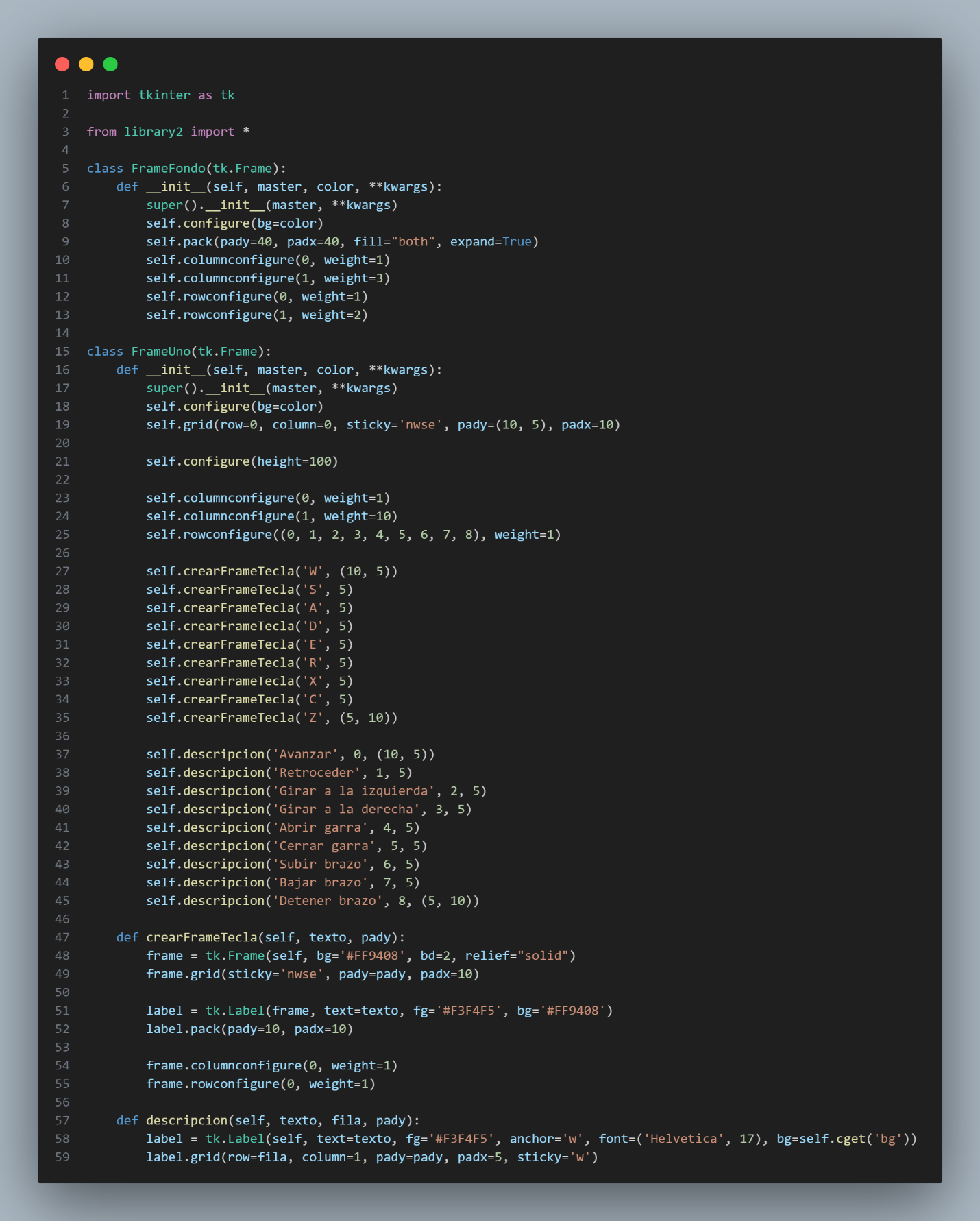
*Energía y trabajo*

Levantar una carga implica realizar trabajo contra la fuerza gravitatoria.

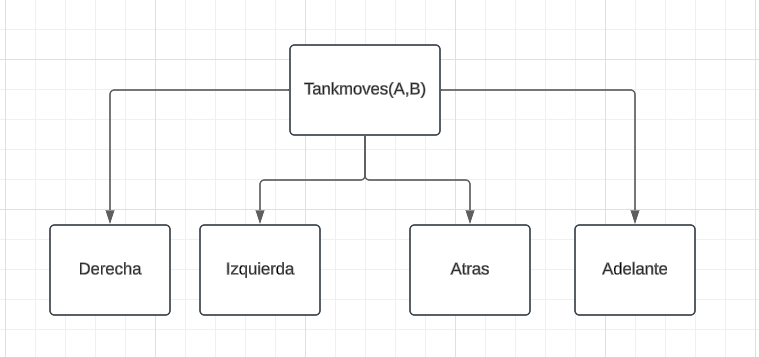
Formula:

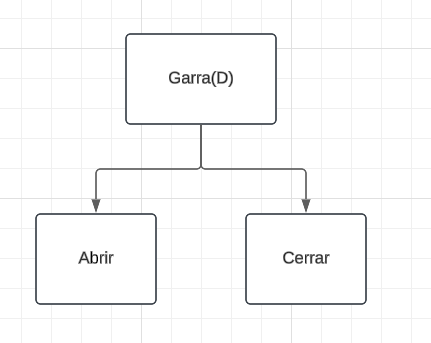
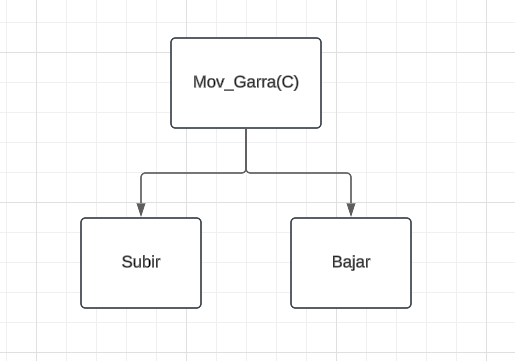
* W: Trabajo realizado
* F: Fuerza aplicada
* d: Distancia recorrida de la carga
* : Angulo entre la fuerza y dirección del movimiento

## Descripción de los programas

* Server:
* Library: 
* Interfaz: 

## Diagramas





# **Resultados**

## Estado Actual del Proyecto

* El robot ya se encuentra construido en su totalidad.
* Cuenta con sus funciones ya definidas tanto como su dirección, movimiento del brazo al igual que de la garra
* Control del robot desde una interfaz gráfica mediante Tkinter y la creación de su servidor
* Wiki del proyecto Actualizada
* Carta Gantt actualizadas

Por lo general todo esta actualizado hasta el día de hoy pero no están terminadas, aún falta proceso y días para llegar a terminar todo, tanto como la wiki como la carta Gantt al igual que terminar todas las bitácoras pendientes de las semanas que quedan como proyecto.

## Problemas Encontrados y Solución Propuesta

|  |  |
| --- | --- |
| **Problemas Encontrados** | **Soluciones** |
| Error de compatibilidad por versiones Se utilizaron funciones de Python no compatibles con la versión instalada en el robot EV3Dev (se usó la versión 3.13, mientras que el robot tiene la 3.5.2). | fue necesario realizar varias correcciones al código para asegurar su compatibilidad. Algunos de los cambios realizados incluyen:   * Modificación de la interfaz: cambio de la biblioteca CustomTKinter a Tkinter. * Ajustes en las funciones de library2.py. * Sustitución del condicional match-case por el condicional if. * Cambio de print con formato (f-string) a print estándar. |
| A la hora de crear la versión 2 del robot tuvimos un problema por la cantidad de piezas que llegaba a usar el robot y un tamaño por lo que se descartó directamente su creación | Crear un diseño nuevo que sea más simple y eficiente para que el robot no tenga un tamaño exagerado y el gasto de piezas no sean tan altas, con lo cual creamos la versión 3 del robot siendo la actual y más eficiente a las anteriores |

# Conclusión

Durante el desarrollo de este proyecto se pudo apreciar bastante dificultad tanto en la creación del robot como los problemas de compatibilidad en su codificación. No obstante, se pudo solucionar todo ese problema llegando a tener un diseño estable en el robot y una interfaz cómoda y sencilla para el usuario.

Por otro lado, igual la calidad del tiempo fue acogedora con los roles dados, también que en mitad de proyecto hicimos un cambio de rol en un integrante donde nos beneficiamos más por las cosas que hacia dando una facilidad por el rol de diseñador.

# Referencias

*Ev3dev home*. (s. f.). ev3dev. <https://www.ev3dev.org/>

Ev3dev. (s. f.). *GitHub - ev3dev/ev3dev-lang-Python: Pure Python bindings for EV3Dev*. GitHub. <https://github.com/ev3dev/ev3dev-lang-python>

Nigel Ward. (2016, 27 octubre). *EV3 Python: Set up an SSH connection from the EV3 to the computer* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=ZfhqZGFJd9A>