

# Proyecto Panzer 2

**Integrantes:** Miguel Fernández  
Edson Galdames  
Lukas Torres  
Bastian Vega

**Asignatura:** Proyecto 1

**Profesor:** Humberto Urrutia



# TABLA DE CONTENIDOS

**01**

**Introduccion**

**02**

**Objetivos**

- Objetivos generales
- Objetivos específicos

**03**

**Planificación del  
proyecto**

- Carta gantt

**04**

**Análisis y diseño**

- Especificación de requerimientos
- Arquitectura
- Interfaz

**05**

**Implementacion**

- Projectiles

# TABLA DE CONTENIDOS

## 06

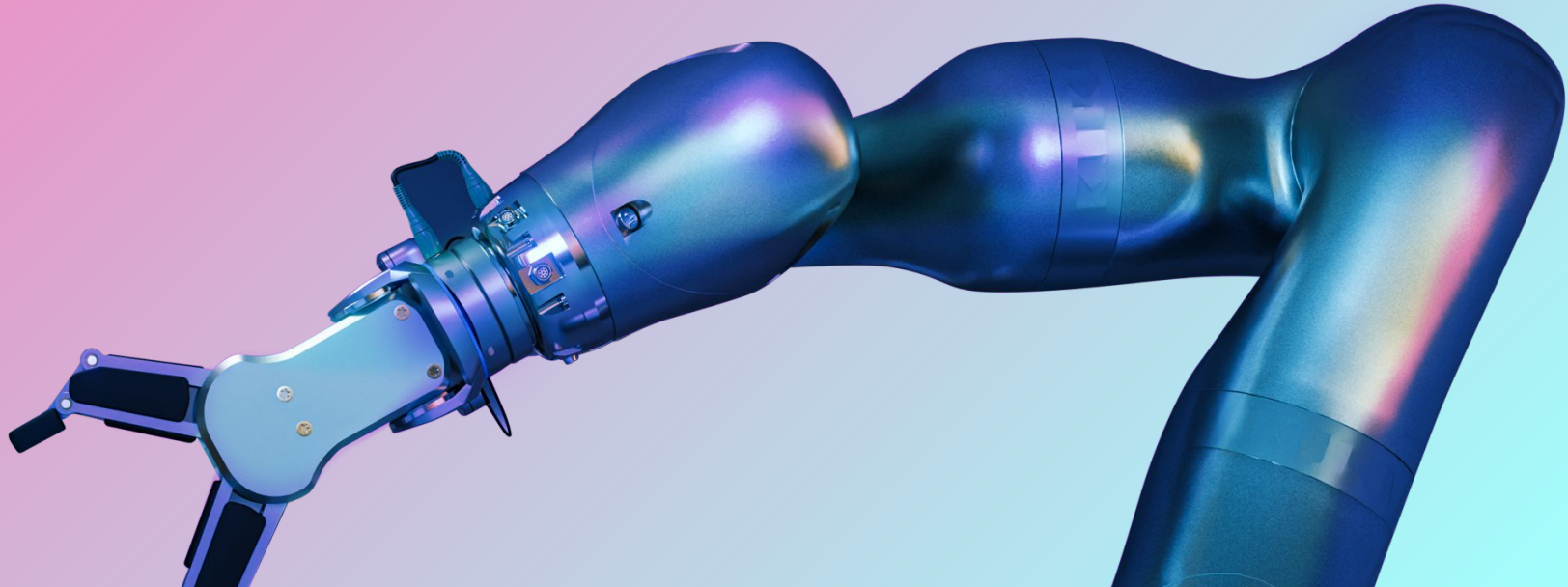
### Resultados

- Estado actual del proyecto
- Problemas encontrados y Soluciones propuestas

## 07

### Conclusión

# 01 Introduccion



# Introduccion

Este proyecto consta en construir un robot con el kit lego mindstorms ev3, el cual tiene que tener movimiento, un cañón y sistema de elevación.







# 02

## Objetivos

# Objetivos general

El objetivo general de este proyecto consta en utilizar el kit de lego MindStorms Ev3 para armar un robot con un cañón que se pueda desplazar, apuntar a un objetivo y disparar.

## Objetivo específico:

- Diseñar al robot para que sea capaz de desplazarse y disparar
- Programar al robot para que cumpla lo anterior
- Desarrollar una interfaz para controlar de manera remota al robot



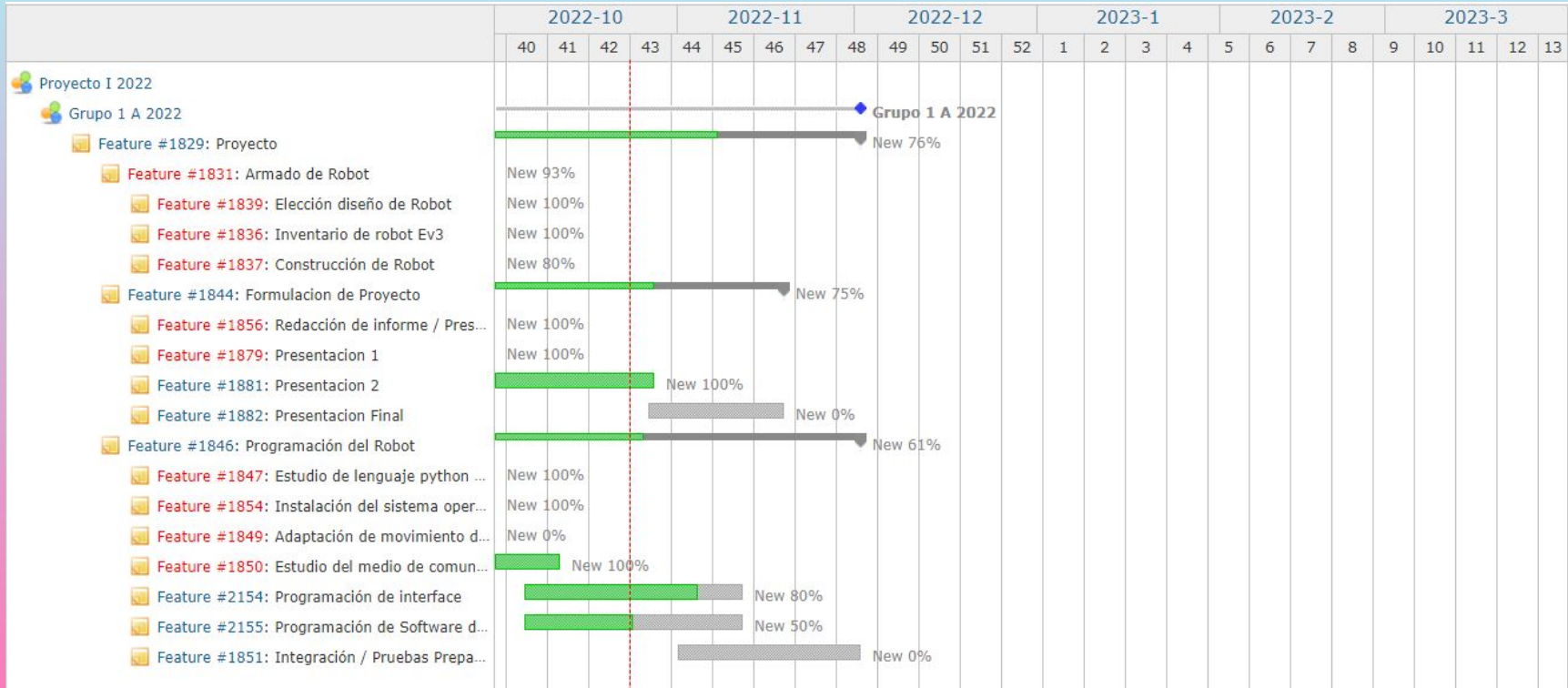
# 03

## Planificación del proyecto





# Carta gantt





# 04 Análisis y diseño

# Especificación de requerimiento

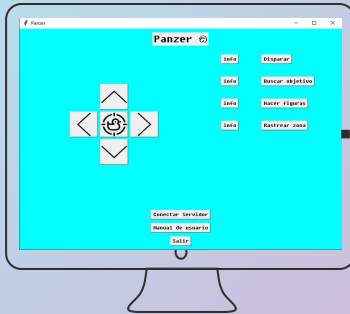
Prioridad:

1. Muy importante
2. Importante
3. No tan importante

Código	Descripción	Prioridad	Revisión	Aceptado	Realizado
1	El robot podrá moverse a voluntad mediante una interfaz gráfica	1	27/10/2022	✓	✓
2	El robot debe tener la posibilidad de disparar y tener un sistema de elevación	2	13/10/2022	✓	✓
3	El código debe estar hecho en python	1	13/09/2022	✓	✓
4	El robot debe estar construido con el kit lego ev3	1	18/09/2022	✓	✓
5	Conexión de la interfaz con el robot mediante SSH	1	25/10/2022	✓	✓

# Arquitectura

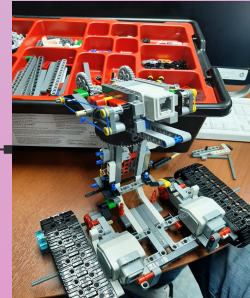
Interfaz grafica



Conexión  
mediante SSH



Interfaz grafica

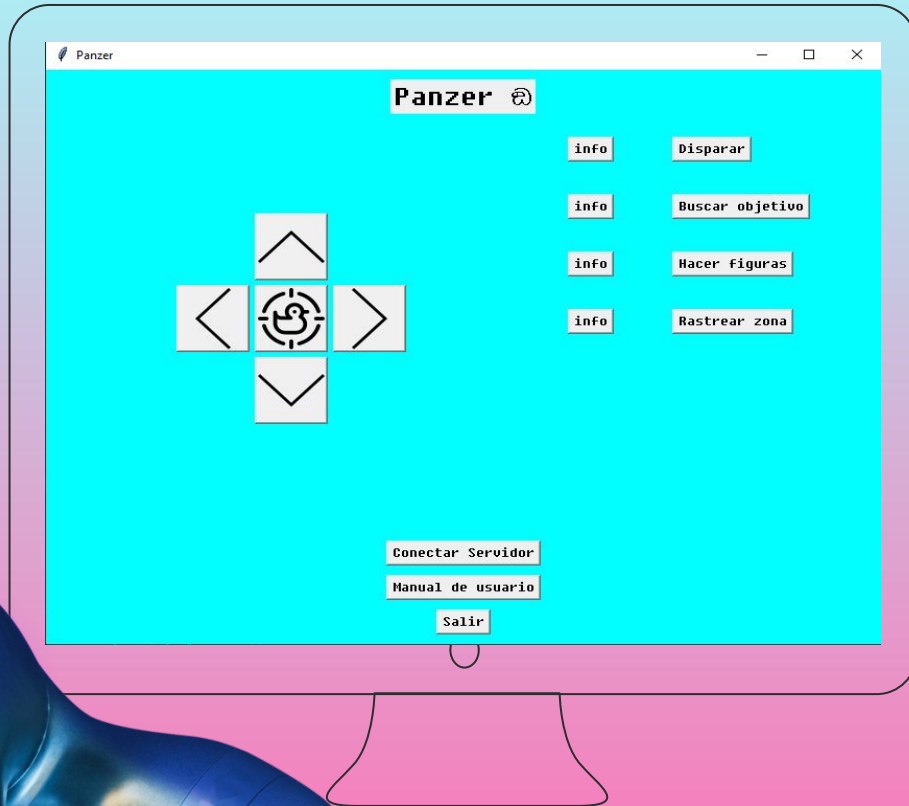


server.py



funciones.py

# Interfaz



La interfaz del robot cuenta con un menú con diversas opciones las cuales nos permiten dar instrucciones al robot desde la interfaz.

- "Disparar" Función que da la instrucción al robot de disparar al objetivo captado por su sensor.
- "Buscar objetivo" Función que da al robot la instrucción de moverse hasta que su sensor capte algún objetivo
- "Hacer figuras" Funciones para que el robot comience a moverse con unos patrones predeterminados con formas de figuras geométricas.
- "Rastrear Zona" Función que da la orden al robot de mapear una zona e identificar los obstáculos de esa Zona.
- "Conectar servidor" Nos permite conectar el robot al servidor que estamos utilizando.
- "Manual de usuario" Es un manual el cual orientará al usuario en el uso del robot y de la interfaz de este.





# 05

## Implementacion

# Fundamento de proyectiles

Para calcular y ejecutar el lanzamiento del proyectil debemos tener presentes dos tipos de movimientos para saber como se comportará el proyectil:

- Movimiento rectilíneo uniforme (M.R.U) en el eje horizontal.

$$x = x_0 + v_x * t$$

$$v_x = v_0 * \cos\theta = cte$$

- Movimiento rectilíneo uniforme acelerado (M.R.U.A) en el eje vertical.

$$y = y_0 + v_{0y} * t + \frac{1}{2} * a_y * t^2$$

$$v_y = v_0 * \sin\theta - g * t$$

# Fundamento de proyectiles

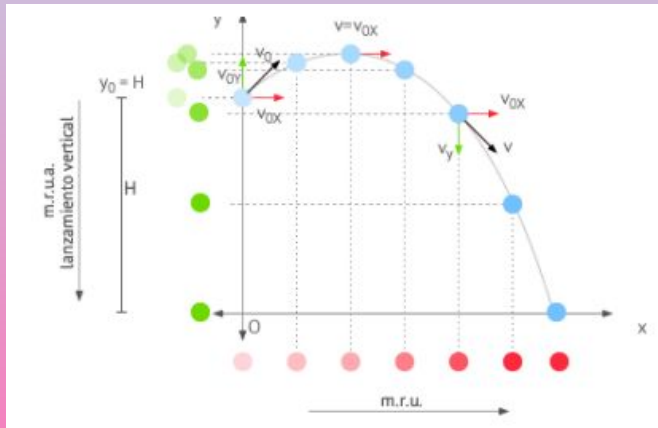
El proyectil al ser lanzado con una velocidad inicial y con un ángulo formado por el eje horizontal provocará que el vector se descomponga en sus componentes horizontales y verticales .

Al tener una velocidad inicial igual a 0 nos permite obtener la altura máxima despejando:

$$h_{\text{máx}} = h_0 + \frac{(v_0 \cdot \sin\theta)^2}{2 \cdot g}$$

También es necesario obtener la distancia máxima del proyectil :

$$\begin{aligned}x &= x_0 + v_0 \cdot \cos\theta \cdot t_{\text{total de vuelo}} \\t_{\text{total de vuelo}} &= 2 \cdot t_{\text{máx}} \\x = R &= x_0 + v_0 \cdot \cos\theta \cdot 2 \cdot \frac{v_0 \cdot \sin\theta}{g} \\R &= x_0 + \frac{v_0^2 \cdot \sin 2(\theta)}{g}\end{aligned}$$



# Descripción de los programas

Con la finalidad de realizar un tiro certero se debe implementar una función que permita al robot calcular distancias y actuar en consecuencia.

Al encontrarse con un objetivo fuera de su rango máximo de tiro (Por estudiar) el robot deberá acercarse a dicho objetivo hasta entrar en un rango de tiro y calcular el ángulo necesario para darle al objetivo visto.





# 06

## Resultados



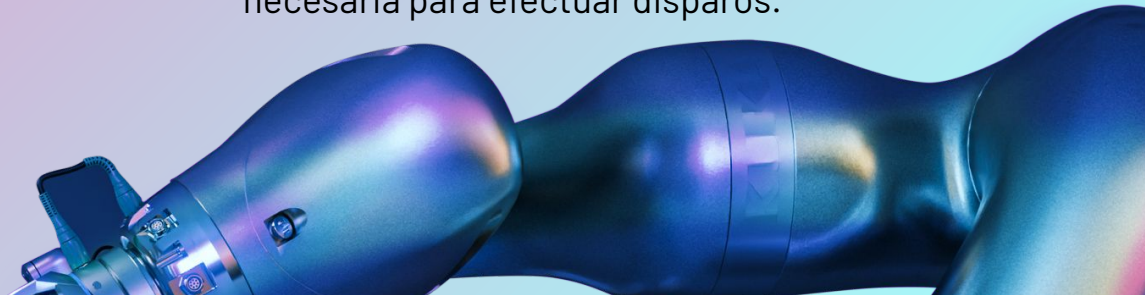
# Estado actual del proyecto

## El proyecto cuenta con:

- Versión final en el armado del robot
- Sistema de disparo listo junto a un sistema de elevación
- Conexión de la interfaz al servidor dentro del robot
- Funciones para moverse

## Actualmente en desarrollo

- Desarrollo de una wiki
- Se está trabajando para mejorar el diseño de la interfaz
- Codificar las funciones que tenemos pensado para nuestro robot
- Algoritmo para calcular la elevación necesaria para efectuar disparos.



# Problemas Encontrados y Solución Propuesta

<b>Problemas encontrados</b>	<b>Soluciones propuestas</b>
Falta de piezas de lego en el armado del sistema elevación.	Adaptar el diseño original para construirlo con las piezas que disponemos.
Limitación técnica del kit EV3 para la implementación de un sistema de rotación y elevación independiente en el cañón era necesario usar dos motores, como el kit ev3 tiene para usar 4 motores simultáneos no se pudo implementar el sistema de rotación .	Utilizar el último puerto de motor disponible para hacer funcionar el sistema de elevación, en cambio para girar el cañón será necesario girar el robot completo.

# Conclusión

# 06

