



PROYECTO

GOLF BOT

Proyecto de elaboración de interfaz para
manejo de GolfBot.

Integrantes:

Sebastián Cáceres
Vaitheare Gómez
Rubén Salas
Kary Tudela
Mario Villalobos

Profesor: Humberto Urrutia
Asignatura: Proyecto I

CONTENIDO

01

Introducción

02

Organización

03

Objetivos

04

Carta Gantt

05

Tabla de costos

06

Evolución del
GolfBot

07

Arquitectura

08

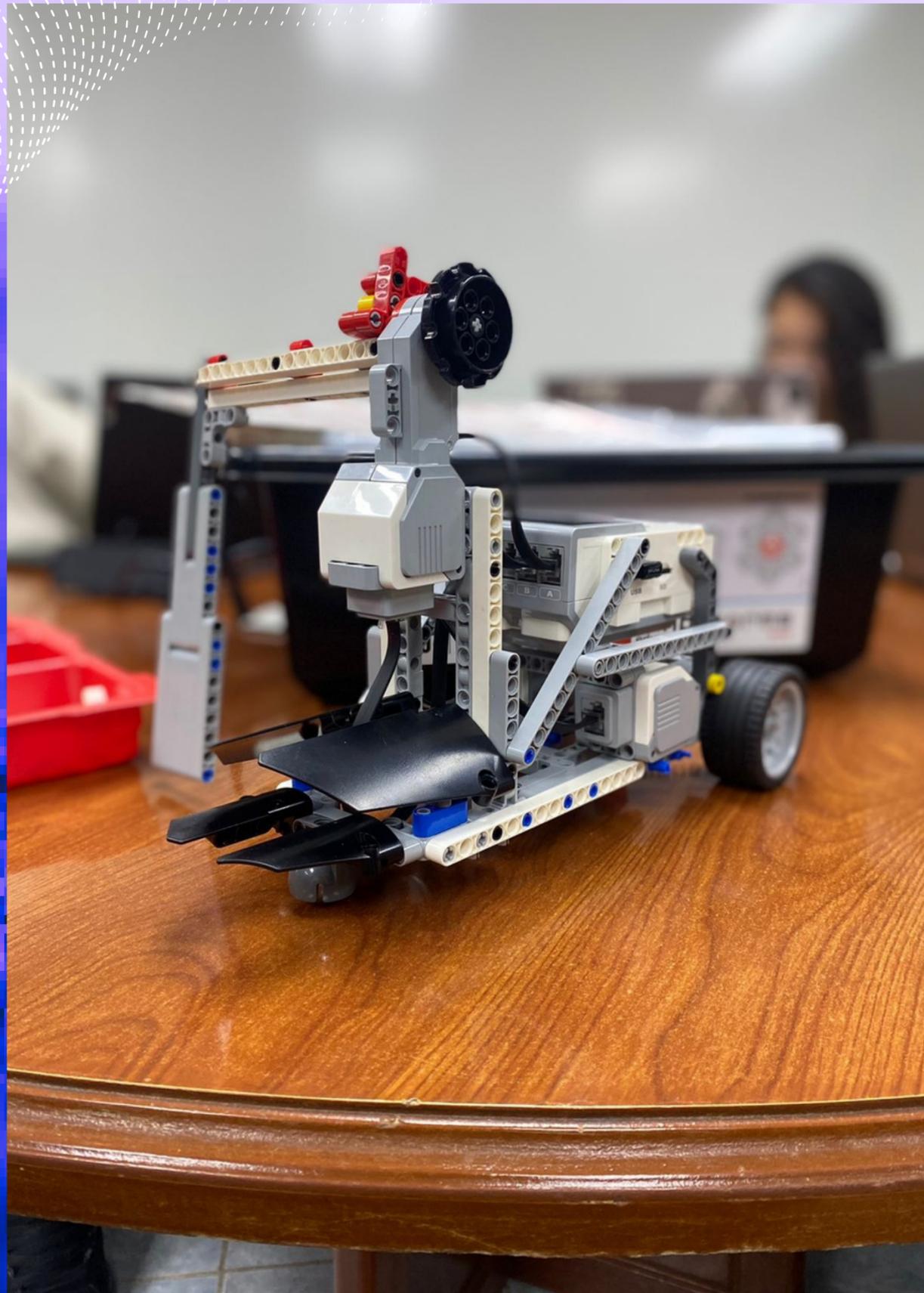
Fundamentos de
Proyectiles

09

Descripción de
programas

10

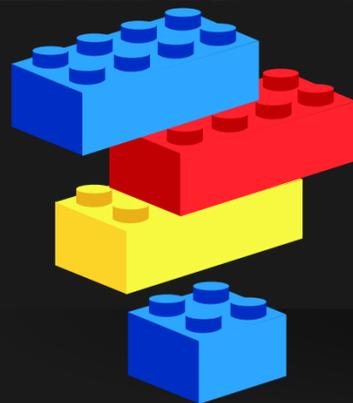
Conclusión



INTRODUCCIÓN AL PROYECTO



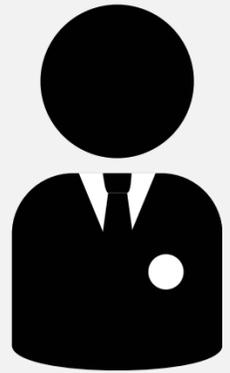
"Golfbot" es un proyecto que desafía a participantes a desarrollar un software para crear un robot. El objetivo es fomentar el trabajo en equipo y el aprendizaje de ingeniería a nivel principiante



La meta concreta es la creación de un robot utilizando Lego Mindstorms, combinando elementos básicos de robótica, como la unión de piezas y la programación interactiva de acciones.

**GOLFBOT**

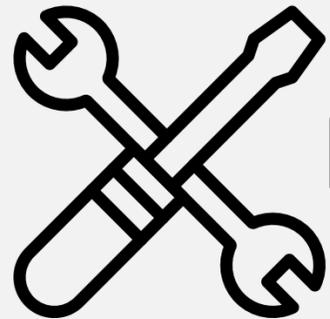
Organización



Jefe de grupo



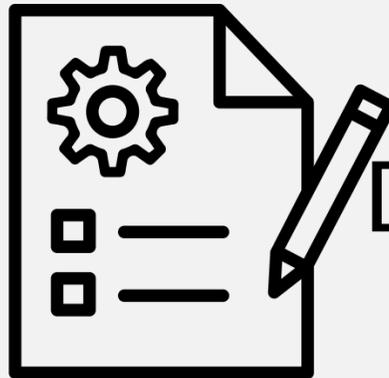
Programador



Ensamblador



Diseñador



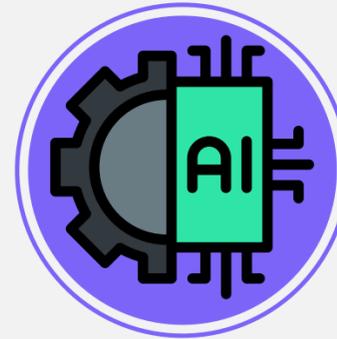
Documentador

OBJETIVO GENERAL Y ESPECÍFICOS



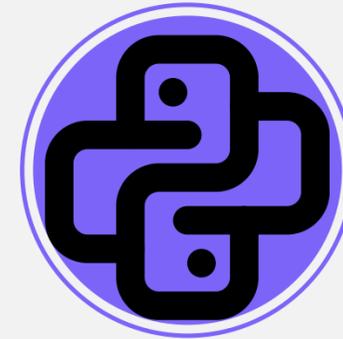
Objetivo General

Construir y programar el
GolfBot



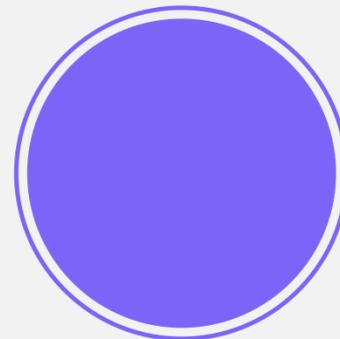
Objetivo Específico

Comunicarse
remotamente con el
Golfbot



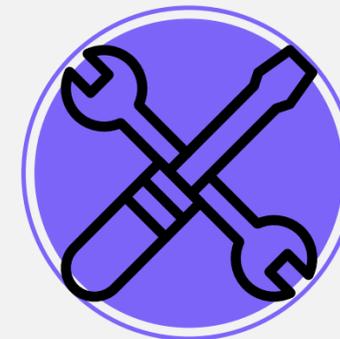
Objetivo Específico

Implementar librerías



Objetivo Específico

Aplicar conocimientos



Objetivo Específico

Ensamblado de piezas



Carta Gantt

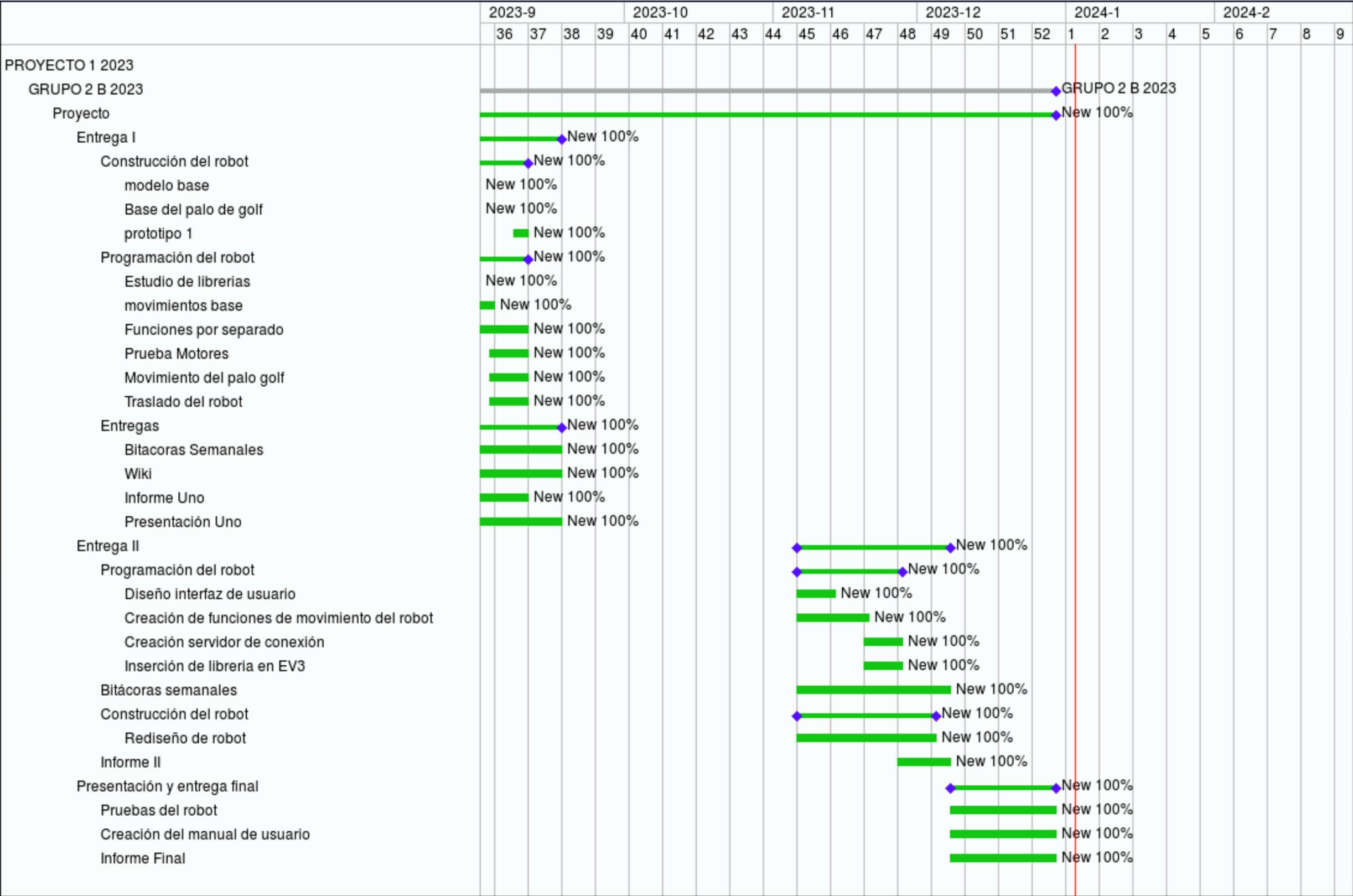


TABLA DE COSTOS

HARDWARE

Producto	Cantidad	Valor	Valor Total
Kit LEGO MINDSTORMS ev3	1	CLP \$1.200.000	CLP \$1.200.000
HP Pavilion Ryzen 5, 8GB X64	1	CLP \$799.000	CLP \$799.000
Toshiba Core i5, 8GB	Renta (31 días por 4 meses)	CLP \$4.650	CLP \$18.600
Tarjeta Memoria Micro SD XC 8GB Kingston	1	CLP \$15.900	CLP \$15.900
Usb Inalámbrico Rojo TP-Link	1	CLP \$17.600	CLP \$17.600
Piezas de repuesto kit	1	CLP \$30.000	CLP \$30.000
		Total	CLP \$2.081.100

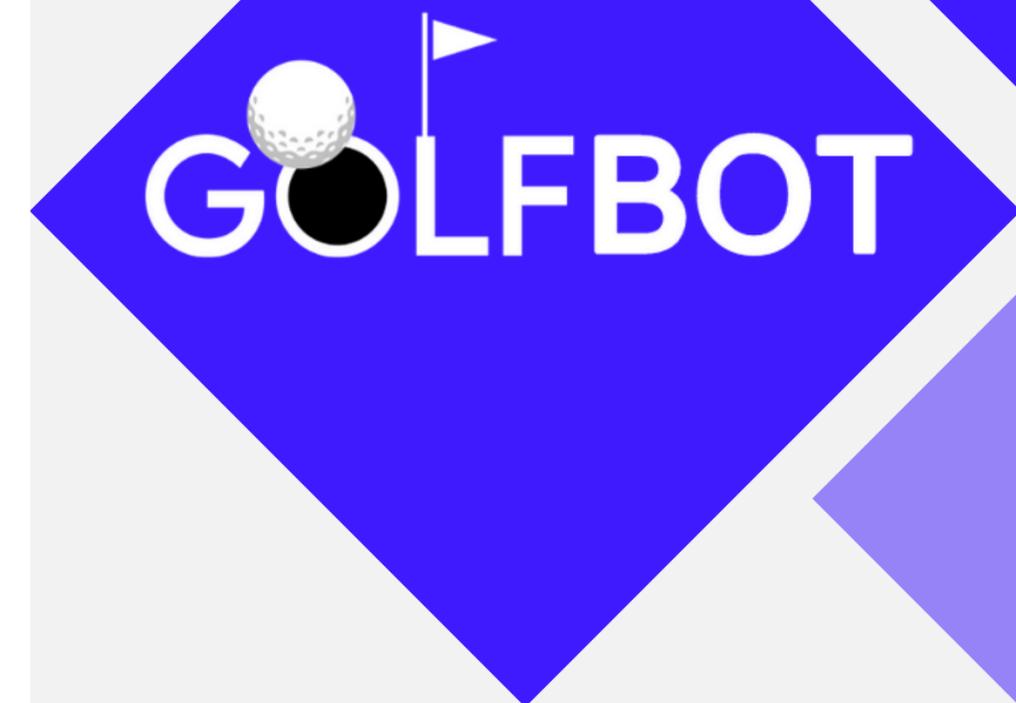


TABLA DE COSTOS DEL PERSONAL



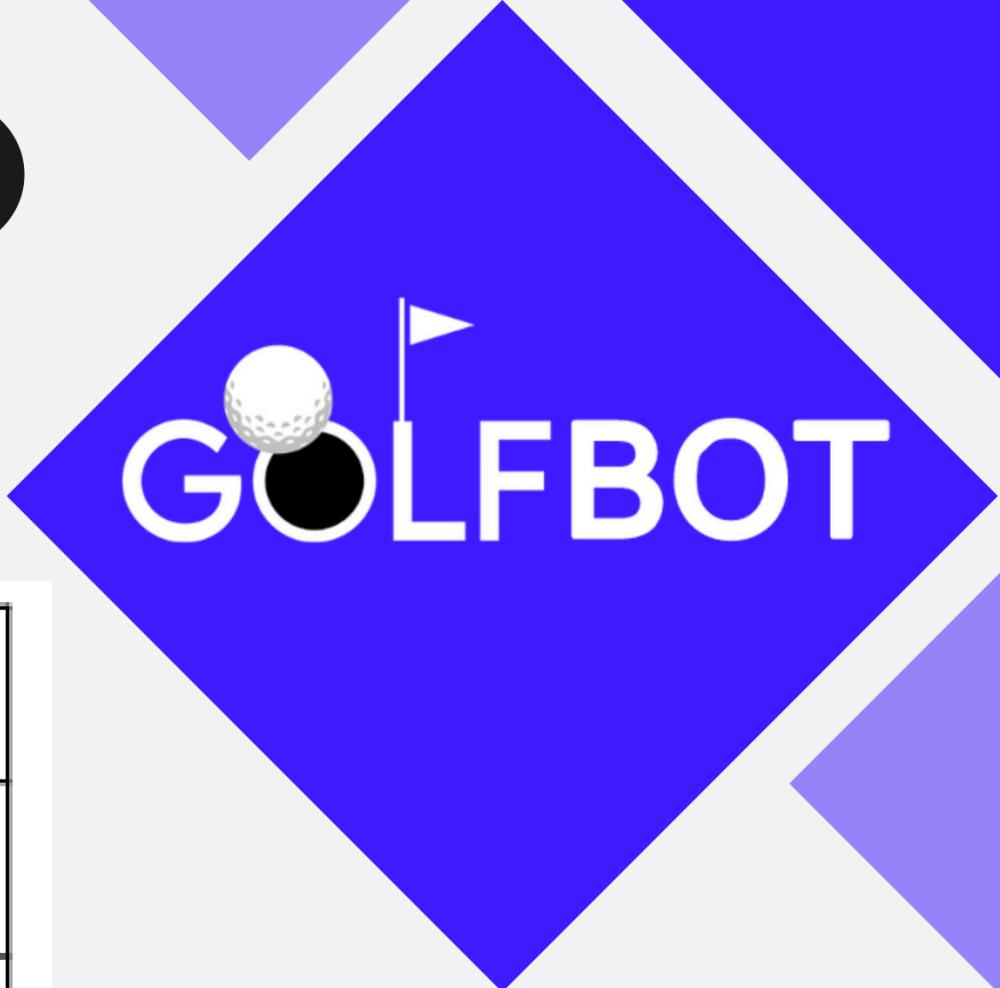
Trabajador	Cargo	Horas Mensuales	Valores Por Hora
Sebastián Cáceres	Ensamblar	15	\$32.000
	Documentar	8.5	\$28.000
	Programar	15	\$35.000
Rubén Salas	Ensamblar	18	\$32.000
	Documentar	15	\$28.000
	Programar	5.5	\$35.000
Mario Villalobos	Ensamblar	15	\$32.000
	Documentar	4.5	\$28.000
	Programar	19	\$35.000
Kary Tudela	Sub Jefe de proyecto	15	\$37.000
	Ensamblar	3.5	\$32.000
	Documentar	5	\$28.000
	Programar	15	\$35.000
Vaitheare Gómez	Jefe de proyecto	20	\$37.000
	Ensamblar	2.5	\$32.000

TABLA DE COSTOS DEL PERSONAL FINAL

Personal	Sueldo mensual	Sueldo por 4 Meses
Sebastián Cacéres	1.243.000	4.972.000
Ruben Salas	1.188.500	4.754.000
Mario Villalobos	1.271.000	5.084.000
Kary Tudela	1.332.000	5.328.000
Vaitheare Gomez	1.345.000	5.380.000
	6.379.500	25.518.000

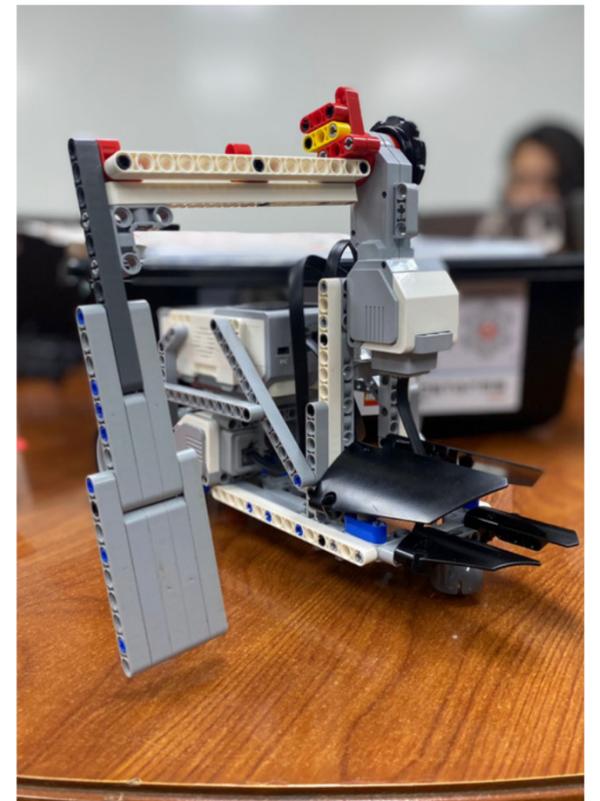
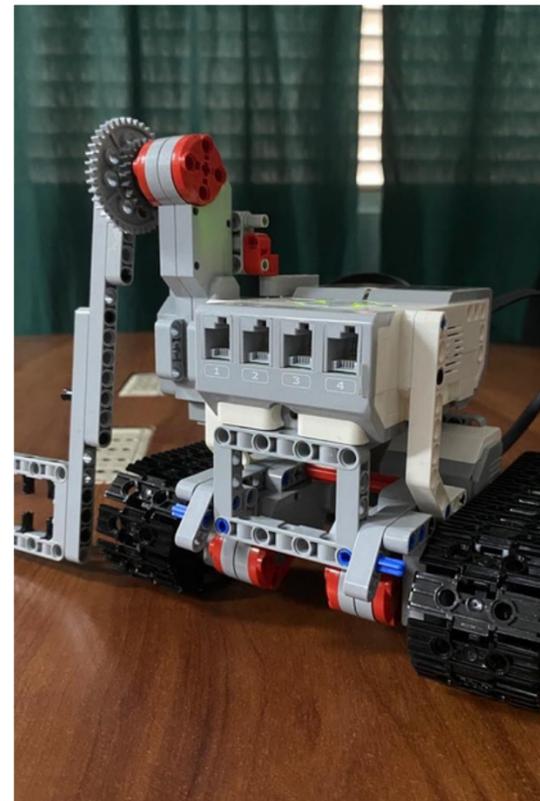


TABLA DE COSTOS FINAL DEL PROYECTO



Costos Hardware	CLP \$2.081.100
Sueldo del personal	CLP \$25.518.000
Total Del Proyecto	CLP \$27.599.000

Evolución del GolfBot



01

02

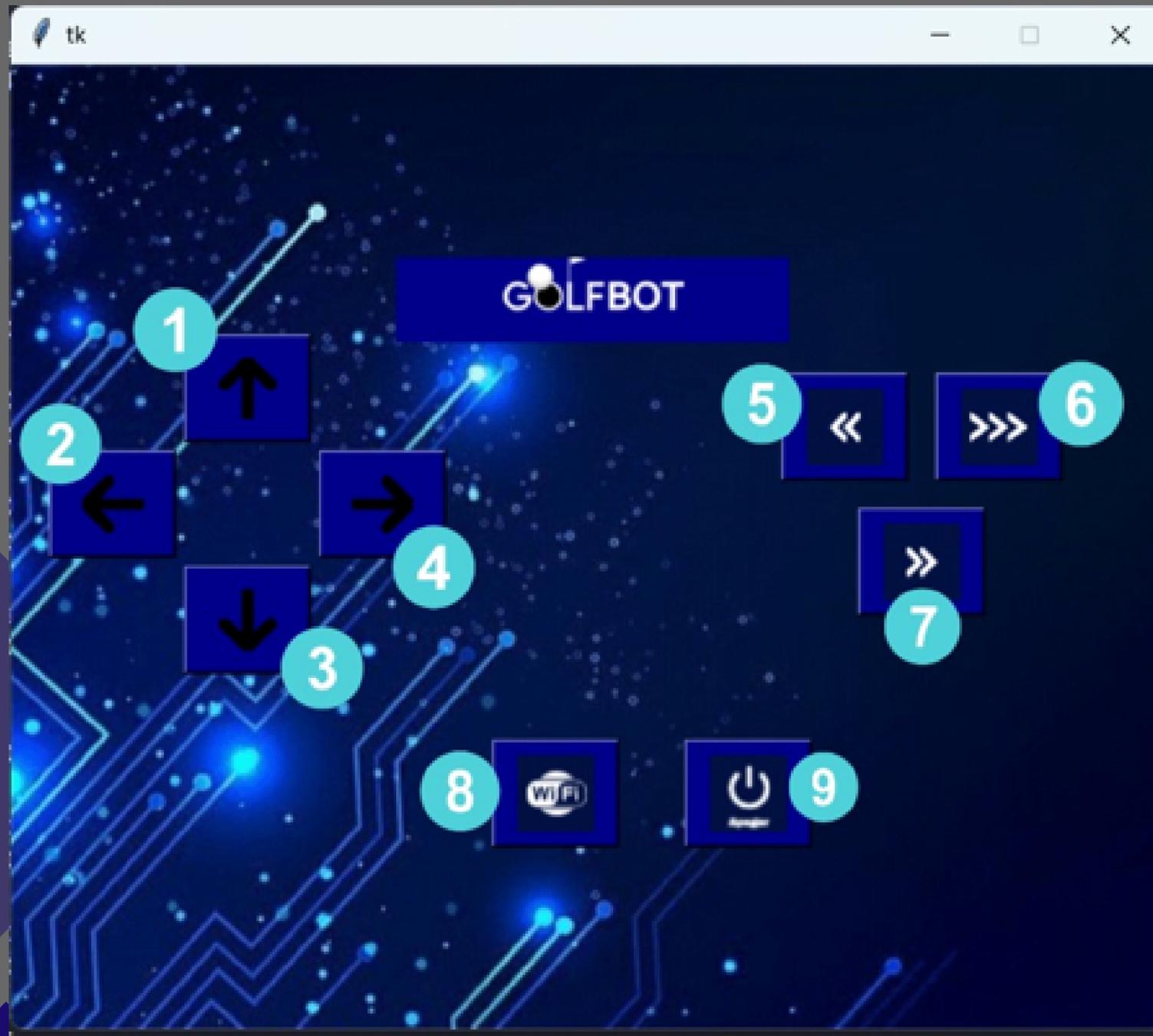
03

04

ARQUITECTURA

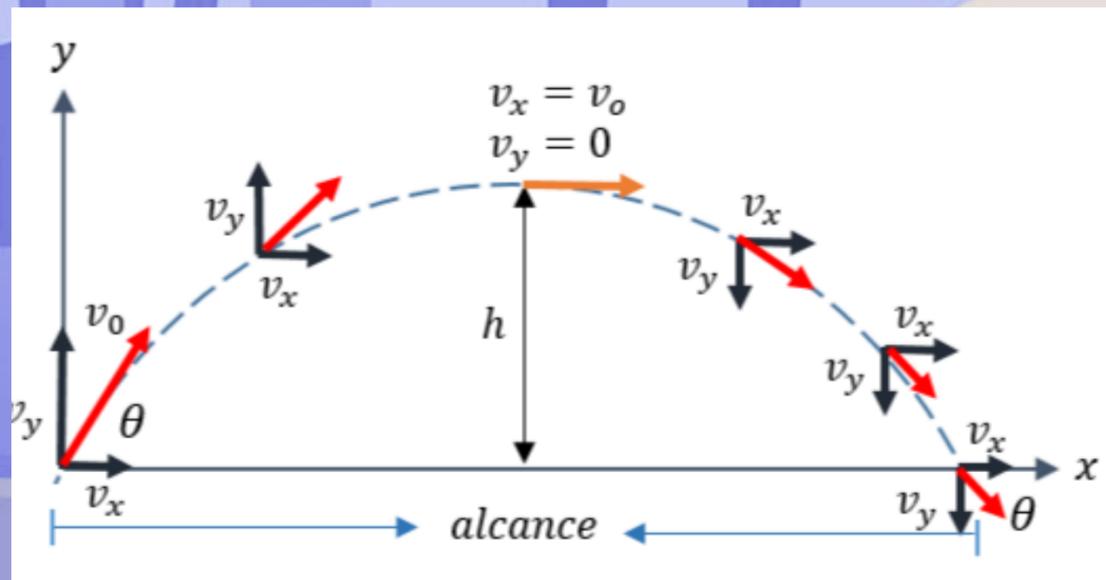


INTERFAZ



- 1.- Arriba
- 2.- Izquierda
- 3.- Abajo
- 4.- Derecha
- 5.- Palo hacia atrás
- 6.- Golpe fuerte del brazo
- 7.- Golpe suave del brazo
- 8.- Conexión con el servidor
- 9.- Cierre de la Interfaz

Fundamento de Projectiles



$$c.- y = y_0 + v_{0y} \times t + \frac{1}{2} \times a \times t^2$$

$$d.- v_y = v_0 \times \text{sen}(\theta) - g \times t$$

Los datos que se conocen son:

$$\theta = 30^\circ$$

$$g = 9.8 \text{ m/s}^2$$

$$x = 0.3 \text{ m} \parallel x_0 = 0 \text{ m}$$

$$y = 0.15 \text{ m} \parallel y_0 = 0 \text{ m}$$

$$v_0 = ? \parallel t = ?$$

De v_0 sacamos que:

$$v_{0x} = v_0 \times \cos(45^\circ) = v_0 \times \frac{\sqrt{2}}{2}$$

$$v_{0y} = v_0 \times \text{sen}(45^\circ) = v_0 \times \frac{\sqrt{2}}{2}$$

Por lo tanto se concluye que:

$$v_{0x} = v_{0y}$$

De la ecuación 'a' se despeja v_0 :

$$x = x_0 + v_{0x} \times t \Rightarrow v_0 = \frac{2x}{t\sqrt{2}}$$

Se reemplaza v_{0y} en la ecuación 'c':

$$y = y_0 + v_{0y} \times t + \frac{1}{2} \times a \times t^2 \Rightarrow 0 = v_0 \times \frac{\sqrt{2}}{2} \times t - 4.9t^2$$

Reemplazamos v_0 :

$$0 = \left(\frac{2x}{t\sqrt{2}}\right) \times \left(\frac{\sqrt{2}}{2}\right) \times t - 4.9t^2$$

Al resolver tenemos que $t \approx 0.2474$ segundos

Y se concluye que $v_0 \approx 1.714 \text{ m/s}$

DESCRIPCIÓN DEL PROGRAMA

Servidor

Es el programa encargado de realizar la conexión mediante un servidor creado por la librería socket de Python.

- Bucle que recibe instrucciones

Librerías

Se definen las funciones responsables de interactuar con el robot

- Acciones que se llevarán a cabo

Interfaz

Presenta una interfaz gráfica desarrollada mediante la librería Tkinter de Python.

- Botones
- Imágenes

Servidor

```
import socket
from library import *
s = socket.socket()
print("Socket creado")
port = 19999
s.bind(("", port))
print("El socket se creo con puerto:{}".format(port))
s.listen(5)
print("EL socket is listening....")
connect, addr = s.accept()
print("Se conecto a {}".format(addr))
while True:
    rawByte = connect.recv(1)
    char = rawByte.decode('utf-8')
    if (char == 'w'):
        avanzar()
    if (char == 's'):
        retroceder()
    if (char == 'd'):
        rotarDerecha()
    if (char == 'a'):
        rotarIzquierda()
    if (char == 'p'):
        golpeSuave()
    if (char == 'f'):
        golpeFuerte()
    if (char == 'i')|
        devolverBrazo()
    if (char == ' '):
        detener()
        detenerBrazo()
```

Libreria

```
from pybricks.hubs import EV3Brick
from pybricks.ev3devices import Motor
from pybricks.parameters import Port
from pybricks.robotics import DriveBase

RobotGolf = EV3Brick
motor_izquierda = Motor(Port.A)
motor_derecha = Motor(Port.B)
motor_brazo = Motor(Port.D)
base_robot = DriveBase(motor_izquierda, motor_derecha, wheel_diameter = 55.5, axle_track = 100)

# Funciones
def avanzar():
    print("Avanzando...")
    base_robot.straight(-10)

def retroceder():
    print("Retrocediendo...")
    base_robot.straight(10)

def rotarDerecha():
    print("Rotando derecha...")
    base_robot.turn(-10)

def rotarIzquierda():
    print("Rotando izquierda...")
    base_robot.turn(10)

def golpeSuave():
    print("Golpe suave...")
    motor_brazo.run(100)

def golpeFuerte():
    print("Golpe fuerte...")
    motor_brazo.run(1000000)

def devolverBrazo():
    print("Retrocediendo brazo...")
    motor_brazo.run(-100)

def detener():
    base_robot.stop()
```

Interfaz

```
from tkinter import *
from tkinter import messagebox
from tkinter import ttk
import socket
import time
import sys

# Funciones -----
def up(event = None):
    clientSocket.send(bytes([ord('w')]))

def down(event = None):
    clientSocket.send(bytes([ord('s')]))

def left(event = None):
    clientSocket.send(bytes([ord('a')]))

def right(event = None):
    clientSocket.send(bytes([ord('d')]))

def StrongArm(event = None):
    clientSocket.send(bytes([ord('f')]))

def WeakArm(event = None):
    clientSocket.send(bytes([ord('p')]))

def BackArm(event = None):
    clientSocket.send(bytes([ord('i')]))

def on_release(event):
    print("click realese")
    clientSocket.send(bytes([ord(' ')]))
```

```
def getAddress():
    w_ip = Tk()
    w_ip.geometry("300x100")
    w_ip.resizable(False, False)
    w_ip.grab_set()
    dato = StringVar(w_ip)
    w_ip.title("Ingreso dirección IP")
    ip = ttk.Entry(w_ip, textvariable=dato).place(x=10, y=10)
    button = Button(w_ip, text = "Conectar", command=lambda:[conectar(dato.get()),w_ip.destroy()]).place(x=170, y=9)
    print(dato.get())

def conectar(adress):
    port = 19999
    try:
        clientSocket.connect((adress,port))
        messagebox.showinfo("Mensaje Servido","Cliente conectado al robot: {0} : {1}".format(adress,port))
    except socket.error:
        messagebox.showwarning("Conexión erronea","No se ha logrado al conexion, verifique la Ip {0}".format(adress))
        clientSocket.close()

# -----

window = Tk()
window.geometry("600x500")
window.resizable(False, False)

# Imágenes -----
Arriba = PhotoImage(file="icono.gif")
Abajo = PhotoImage(file="iconoabajo.gif")
Derecha = PhotoImage(file="iconoderecha.gif")
Izquierda = PhotoImage(file="iconoizquierda.gif")
Fondo = PhotoImage(file="fondo.gif")
Wifi = PhotoImage(file="wifi.gif")
Mas_Rapido = PhotoImage(file="muyrapido.gif")
Rapido = PhotoImage(file="rapido.gif")
Logo = PhotoImage(file="LogoGolf2.png")
Atras = PhotoImage(file="rapido2.png")
Apagar = PhotoImage(file="Apagar.gif")

#-----
```

Interfaz

```
fondo = Label(window, width=1500, height=1000, image=Fondo)
fondo.pack()

Golf = Label(window, width=200, height=40, image=Logo, bg="blue4").place(x=200,y=100)

# Botón Avanzar
boton_avanzar = Button(window, repeatdelay = 50, repeatinterval = 50, image = Arriba,
                       command = up, height = 50, width = 60, bg = "blue4")
boton_avanzar.place(x=90,y=140)
boton_avanzar.bind('<ButtonRelease-1>',on_realese)

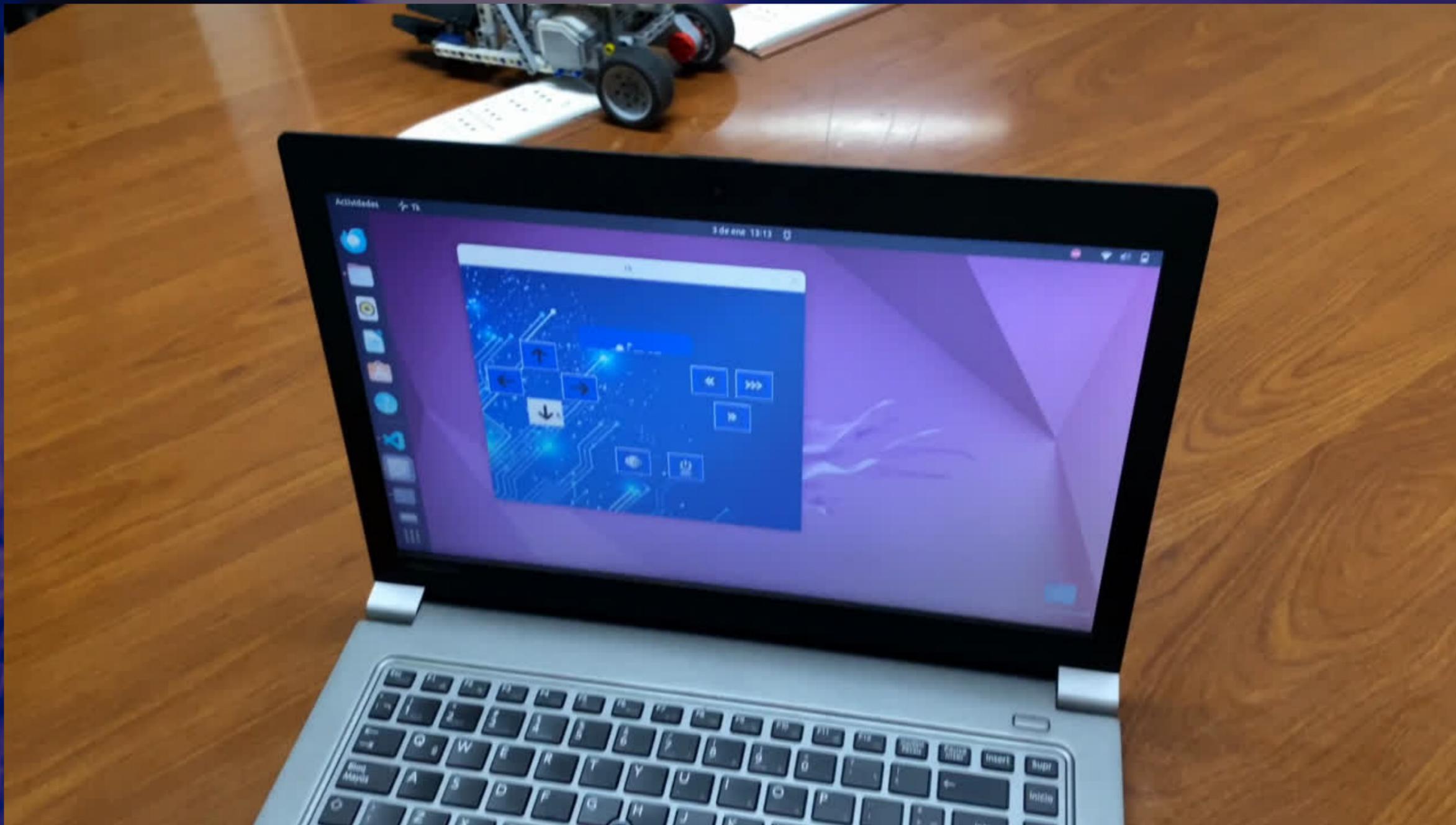
# Botón Retroceder
boton_retroceder = Button(window, repeatdelay = 50, repeatinterval = 50, image = Abajo,
                          command = down, height = 50, width = 60, bg = "blue4")
boton_retroceder.place(x=90,y=260)
boton_retroceder.bind('<ButtonRelease-1>',on_realese)

# Botón derecha
boton_derecha = Button(window, repeatdelay = 50, repeatinterval = 50, image = Derecha,
                      command = right, height = 50, width = 60, bg = "blue4")
boton_derecha.place(x=160,y=200)
boton_derecha.bind('<ButtonRelease-1>',on_realese)

# Botón Izquierda
boton_izquierda = Button(window, repeatdelay = 50, repeatinterval = 50, image = Izquierda,
                         command = left, height = 50, width = 60, bg = "blue4")
boton_izquierda.place(x=20,y=200)
boton_izquierda.bind('<ButtonRelease-1>',on_realese)

# Botón Lanzar
boton_lanzar = Button(window, repeatdelay = 50, repeatinterval = 50, image = Rapido,
                     command = WeakArm, height = 50, width = 60, bg = "blue4")
boton_lanzar.place(x=440,y=230)
boton_lanzar.bind('<ButtonRelease-1>',on_realese)
```

Muestra del GolfBot



CONCLUSIÓN

- El robot cumple con los objetivos iniciales y de planificación.
- Se cumplió con la fecha límite.

Para concluir con nuestra presentación podemos afirmar que este proyecto nos ha resultado de mucho apoyo para desarrollar tanto habilidades individuales como grupales.