

**UNIVERSIDAD DE TARAPACÁ**



**FACULTAD DE INGENIERÍA**



**DEPARTAMENTO DE INGENIERÍA EN COMPUTACIÓN E INFORMÁTICA**



**Manual de usuario: “Heracross”**

**Alumno(s):**

- Hilda Albarrecin
- Guillermo Pino
- Martín Gonzales
- Isaac Contreras

**Asignatura:** Proyecto 1

**Profesor:** Humberto Urrutia López

# Índice de Contenido

|   |   |
|---|---|
| 1. Historial de cambios.....                    | 3 |
| 2. Introducción .....                           | 4 |
| 3. Descripción del robot y sus operaciones..... | 4 |
| 4. Requerimientos mínimos .....                 | 4 |
| 5. Procedimientos .....                         | 5 |
| 5.1. Instalación.....                           | 5 |
| 5.2 Obtención del software .....                | 5 |
| 5.3 Uso del software.....                       | 6 |
| 6. Posibles errores y sus soluciones.....       | 7 |
| 7. Referencias.....                             | 7 |

## 1. Historial de Cambios

|             |                                   |
|-------------|-----------------------------------|
| Versión     | 1.0                               |
| Descripción | Realización del manual de usuario |
| Autor       | Isaac Contreras                   |
| Fecha       | 12/12/2024                        |

|             |                     |
|-------------|---------------------|
| Versión     | 2.0                 |
| Descripción | Corrección de error |
| Autor       | Isaac Contreras     |
| Fecha       | 13/12/2024          |

## 2. Introducción

Este manual de usuario se ha redactado con la intención de guiar al usuario en la preparación, instalación y uso correcto del robot “Heracross” del proyecto “Team heracross”.

## 3. Descripción del robot y sus operaciones

El robot fue diseñado y construido de forma que consiga el objetivo de recoger, trasladar y soltar una pelota de ping pong, para esto el robot está equipado con ruedas y una garra funcionales.

Acciones del robot:

- Movimiento rectilíneo: El robot es capaz de desplazarse hacia adelante o atrás.
- Rotación: El robot es capaz de rotar en ambas direcciones.
- Operación de la garra: El robot está equipado con una garra que es capaz que se puede abrir, cerrar, subir o bajar.

## 4. Requerimientos mínimos

Para el funcionamiento correcto del robot, se requieren hardware y software listados a continuación:

Software:

- Sistema operativo: Windows o Linux
- Lenguaje de programación: Python 3.11 o superior
- IDE: Visual Studio Code
- Extensiones: Lego Mindstorm EV3 para Visual Studio Code
- Librerías: Pybricks, Sockets.

Hardware:

- Conexión: Wi-fi, adaptador Wi-fi USB
- Ordenador con, al menos:

- Procesador: 2 GHz
- Memoria RAM: 1GB
- Espacio de almacenamiento disponible.

## 5. Procedimientos

### 5.1 Instalación:

1. Insertar el adaptador WI-FI USB en el puerto correspondiente en el EV3 Brick
2. Iniciar Visual Studio Code, crear una carpeta de proyecto EV3 e insertar los archivos EV3server.py e Interfaz.py en ella.
3. Conectar el ordenador y el EV3 Brick a la misma red WI-FI y luego establecer la conexión en Visual Studio Code.
4. Verificar los valores de la variable “ev3\_ip” en Interfaz.py, este valor debe igualar al de la esquina superior izquierda del menú principal del EV3 Brick.
5. Actualizar la variable “port” en ambos códigos por uno disponible. Para esto, use el terminal de Visual Studio Code, con el comando:  

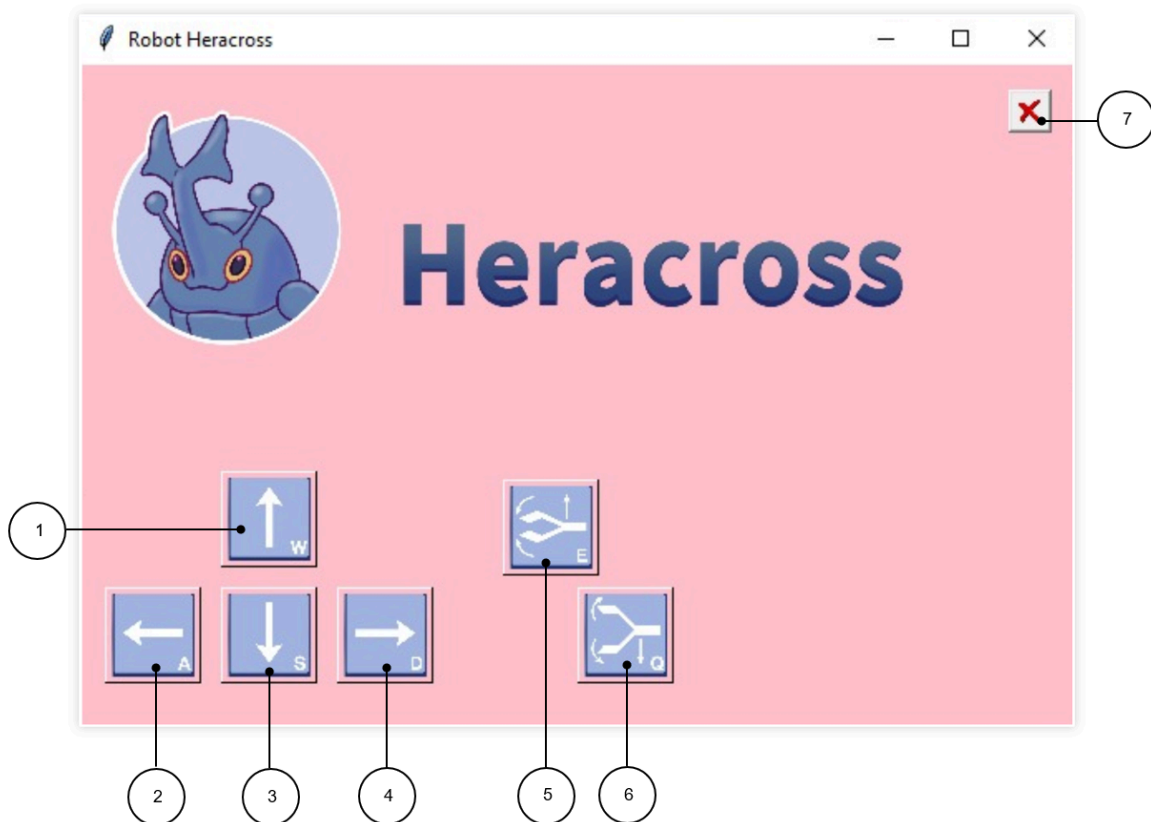
```
Get-NetTCPConnection | Where-Object { $.State -eq 'Listen' },
```

busque un puerto en el que diga “Listening” y “0.0.0.0”, el valor de “port” debe ser el puerto asociado.
6. Ejecutar ambos códigos.

### 5.2 Obtención del software:

- Python: Descargar e instalar python en <https://www.python.org/downloads/>
- Visual Studio Code: Descargar e instalar en <https://code.visualstudio.com/download>
- LEGO Mindstorm EV3: Buscar esta misma extensión en las extensiones de Visual Studio Code.
- Pybricks: Ejecutar “pip install pybricks” en el terminal de Visual Studio Code, una vez conectado al robot.
- Sockets: Ejecutar “pip install sockets” en el terminal de Visual Studio Code, una vez conectado al robot.
- Tkinter: Ejecutar “pip install tk” en el terminal de Visual Studio Code, una vez conectado al robot.

### 5.3 Uso del software:



1. Botón de movimiento hacia adelante.
2. Botón de giro hacia la izquierda.
3. Botón de movimiento hacia atrás.
4. Botón de giro hacia la derecha.
5. Botón para cerrar y luego subir la garra.
6. Botón para bajar y luego abrir la garra.
7. Botón para cerrar el programa.

## 6. Posibles errores y sus soluciones.

- Dirección IP: La dirección ip es incorrecta, se debe verificar y corregir.
- Puerto: El puerto ya no está disponible, se debe verificar y corregir.
- Conexión: Se ha desconectado el EV3 Brick o el ordenador a la red WI-FI, o esta misma dejó de funcionar. Se debe revisar las conexiones y encontrar la desconexión.
- Batería: El EV3 Brick o el ordenador se han descargado. Identificar el dispositivo descargado y cargarlo.

## 7. Referencias

Documentación de Pybricks: <https://pybricks.com/ev3-micropython/>

Descarga Python: <https://www.python.org/downloads/>

Descarga Visual Studio Code <https://code.visualstudio.com/download>