Universidad Nacional de La Matanza

Sistemas Operativos Avanzados

**Año 2019**

**Trabajo Practico**

**“LaberintoSmart”**

**Integrantes Equipo de Proyecto**

|  |  |  |
| --- | --- | --- |
| D.N.I. | Nombre | E-Mail |
| 39.336.665 | Focaraccio, Ezequiel | eafocaraccio@gmail.com |
| 39.923.347 | Galluzzo, Luciano | luchogalluzzo421@gmail.com |
| 40.378.661 | Agustín, Riva | aagusriva@gmail.com |

**Profesores:**

* Graciela De Luca (Jefa de Catedra)
* Waldo A. Valiente
* Sebastián Barillaro
* Mariano Volker
* Carnuccio, Esteban Andrés
* Gerardo García

Índice

[Introducción 5](#_Toc12309752)

[Descripción del proyecto 6](#_Toc12309753)

[Auto-Robot creado con Arduino 6](#_Toc12309754)

[Aplicación Android 10](#_Toc12309755)

[Inicio 10](#_Toc12309756)

[Resolviendo 12](#_Toc12309758)

[Registros 14](#_Toc12309759)

[Información 15](#_Toc12309760)

[Componente HW utilizado 16](#_Toc12309761)

[Microcontroladora 16](#_Toc12309762)

[Arduino UNO 16](#_Toc12309763)

[Sensores 16](#_Toc12309764)

[Sensor Óptico Reflectivo Infrarrojo Cny70 16](#_Toc12309765)

[Sensor Ultrasonido Hc-sr04 Distancia 16](#_Toc12309766)

[Modulo Bluetooth Hc06 Uart Ttl Esclavo 17](#_Toc12309767)

[Actuadores 17](#_Toc12309768)

[Motor Dc 3v A 6v Caja Reductora 17](#_Toc12309769)

[Doble Puente H Driver L298n 17](#_Toc12309770)

[Modulo Bluetooth Hc06 Uart Ttl Esclavo 17](#_Toc12309771)

[Diagramas 18](#_Toc12309772)

[Diagrama de Estados 18](#_Toc12309773)

[Diagrama de Software 19](#_Toc12309775)

[Diagrama Físico 19](#_Toc12309776)

[Diagrama Funcional 20](#_Toc12309777)

[Diagrama Lógico 20](#_Toc12309778)

[Anexo – Código Arduino 21](#_Toc12309779)

# Introducción

LaberintoSmart es un proyecto que nace de la idea de los alumnos con la curiosidad de ver cómo funcionan los seguidores de líneas. A partir de eso, y luego de varias discusiones, se llegó al común acuerdo de agregarle una distinción para que pueda resolver laberintos.

El proyecto está dividido en dos partes: una aplicación Android y un robot formado con un microcontrolador Arduino.

Por un lado, el robot es un auto a escala utilizando ciertos sensores específicos para lograr el objetivo. Dicho dispositivo será capaz de, como bien dijimos anteriormente, resolver un laberinto construido con líneas negras sobre una base blanca.

Además, el coche será capaz de evitar choques frente a objetos que aparezcan en el lado posterior del mismo gracias a un sensor ultrasónico capaz de medir distancias específicas.

Por otro lado, tendremos la posibilidad de realizar diferentes funcionalidades desde una aplicación en sistema Android. Dicha aplicación nos permitirá desde encender el robot hasta tener un seguimiento detallado del historial de laberintos realizados.

# Descripción del proyecto

Como bien hemos mencionado en la introducción, LaberintoSmart es un proyecto que consiste en 2 partes bien definidas: un robot formado con un microcontrolador Arduino y una aplicación Android.

A continuación, entraremos en más detalle sobre cada uno de estos.

## Auto-Robot creado con Arduino

La primera parte del proyecto consiste en un auto a escala creado en base a un microcontrolador Arduino UNO capaz de moverse de forma completamente autónoma guiándose a través de líneas negras sobre un fondo negro, a lo que comúnmente se llama “Seguidor de líneas”.

Este seguidor de líneas será capaz de también resolver cualquier laberinto que se le presente a través de la regla de la mano izquierda (dicha regla especifica que, para resolver un laberinto, siempre se debe dejar el brazo izquierdo pegado a la pared, dándole más prioridad doblar a la izquierda que cualquier otra cosa).

Además, el auto ofrece la posibilidad de, una vez que haya resuelto el laberinto, optimizarlo para encontrar el camino más corto hasta la salida, sin la necesidad de tener que realizar “caminos muertos” o “giros en U”.

Como si esto fuera poco, el robot permite de un modo “Manual” en el que en cada intersección que encuentre en el camino, este se quede esperando por la indicación del ser humano para dónde dirigirse a través de la aplicación móvil.

Al finalizar el modo manual, otorgamos la posibilidad de que el robot demuestre que aprendió el camino exactamente como le fue indicando por el ser humano. Para esto, al igual que en la optimización, se deberá colocar al robot en el mismo lugar de partida y este realizará el mismo camino que le fue indicado anteriormente.

Por último, el auto posee la capacidad de detenerse frente a objetos que surjan en el camino del laberinto para no chocar con estos, dando aviso mediante la aplicación.

**Imagen que contiene objeto

Descripción generada automáticamente**

**Imagen que contiene electrónica

Descripción generada automáticamente**

**Imagen que contiene electrónica

Descripción generada automáticamente**

**Imagen que contiene interior, suelo, mesa

Descripción generada automáticamente**

**Imagen que contiene electrónica

Descripción generada automáticamente**

## Aplicación Android

Esta segunda parte del proyecto consiste en una aplicación móvil que nos sirve de ayuda para ejecutar y poner en práctica todas las funcionalidades del robot. Además, ofrece la posibilidad de llevar un registro de todas esas ejecuciones dentro de un historial para luego observar los tiempos y el estado de cada una.

### Inicio

### Imagen que contiene captura de pantalla Descripción generada automáticamente

#### Comprobar Conexión

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Dentro de esta sección se podrá crear una conexión con el Bluetooth del robot realizando una búsqueda en caso de que todavía no se encuentre emparejado nuestro celular con el dispositivo.

Para ello, debemos presionar en el botón “Comprobar Conexión” y esperar a que se nos abra una nueva pantalla en donde se observaran otros 3 botones:

1. *Activar/Desactivar*: Aquí podremos encender y apagar la conexión bluetooth de nuestro celular.
2. *Ver dispositivos emparejados*: Aquí podremos observar aquellos dispositivos que ya se encuentran emparejados a nuestro celular.
3. *Buscar dispositivos*: Aquí podremos buscar nuestro HC-06 con el nombre de “LSmart\_BT” e ingresar la clave “4321”.

#### Iniciar

También, en la pantalla de inicio, se podrá dar inicio al robot para la resolución del laberinto. En el momento en que se presione el botón “Iniciar”, aparecerá un pop-up preguntando si deseamos ejecutar el modo Automático o el modo Manual.

Luego de elegir una de las opciones, automáticamente se conectará a nuestro celular con el bluetooth del robot y nos llevará a la sección de “Resolviendo”.

### Imagen que contiene captura de pantalla Descripción generada automáticamenteResolviendo

En esta pantalla nos mostrará el estado del robot en el que se encuentra en la resolución del laberinto y nos otorgará la posibilidad de finalizarlo en cualquier momento. Además, mostrará un cronometro con el tiempo que va tardando el robot en resolver el laberinto.

En esta actividad, también tenemos la posibilidad de encender un led si es que el sensor de Luz del celular detecta oscuridad y de frenar momentáneamente al robot si es que el sensor de Distancia del celular detecta que esta siendo tapado.

#### Modo Manual

Si el modo elegido fue el modo Manual, cuando el robot detecte una intersección se frenará esperando a que nosotros le indiquemos que decisión tomar. Esto lo haremos con el sensor Acelerómetro del celular, girando este para el lugar que queremos que doble. Es decir, si queremos que el robot doble a la derecha, debemos inclinar nuestro celular 90° a la derecha y viceversa. En cambio, si queremos que siga adelante, debemos inclinar el celular para adelante de forma que la pantalla quede boca arriba.

Para que esto funcione correctamente, el celular debe encontrarse en la posición por defecto (perpendicular al piso) antes de ejecutar el modo manual y mantenerlo así durante toda la ejecución.

Una vez que el robot haya resuelto el laberinto, nos mostrará una pantalla con el tiempo que tardo y el estado con el que finalizo. Además, nos ofrecerá la posibilidad de pedirle al robot que vuelva a recorrer el laberinto ejecutando el mismo camino que nosotros le dijimos que haga.

#### Modo Automático

Si el modo que elegimos fue el modo Automático, el robot resolverá el laberinto de forma autónoma hasta encontrar el fin.

Una vez que esto ocurra, nos mostrará una pantalla con el tiempo que tardo y el estado con el que finalizo. Además, nos ofrecerá la posibilidad de pedirle al robot que vuelva a recorrer el laberinto de manera optimizada, es decir, con el camino más corto aprendido durante la ejecución anterior.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

### Registros

En esta sección se podrá visualizar todos los registros que hemos realizado en nuestro celular con el robot de LaberintoSmart. En él se podrá ver la fecha y la hora en la que fue ejecutado, el estado con el que finalizó y el tiempo en segundos que tardó.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

### Información

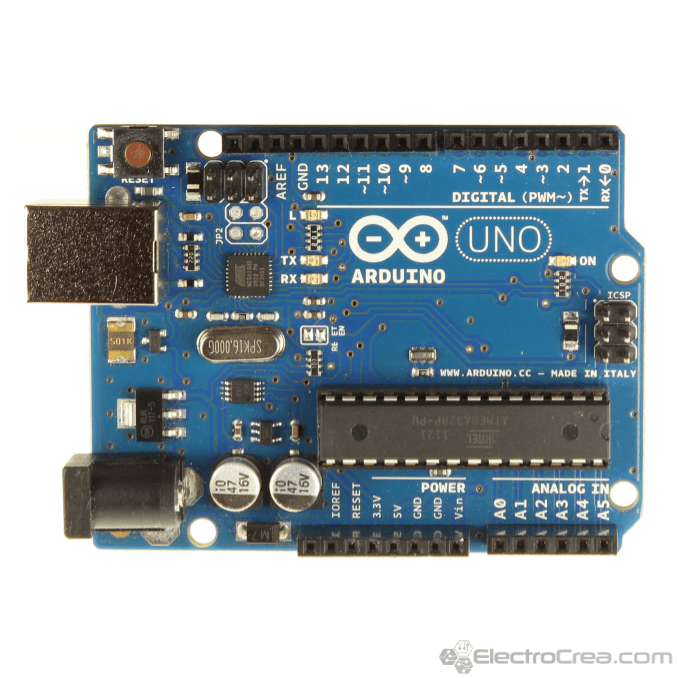
En esta sección se podrá encontrar los integrantes del desarrollo de este proyecto.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

# Componente HW utilizado

## Microcontroladora



### Arduino UNO

Es la placa mediante la cual se procesará todo el código junto con la lógica para la correcta resolución del laberinto, tanto en el modo Manual como en el modo Automático. Además, en ella se procesará el algoritmo de optimización para obtener el camino mas corto hasta la salida.

## Sensores



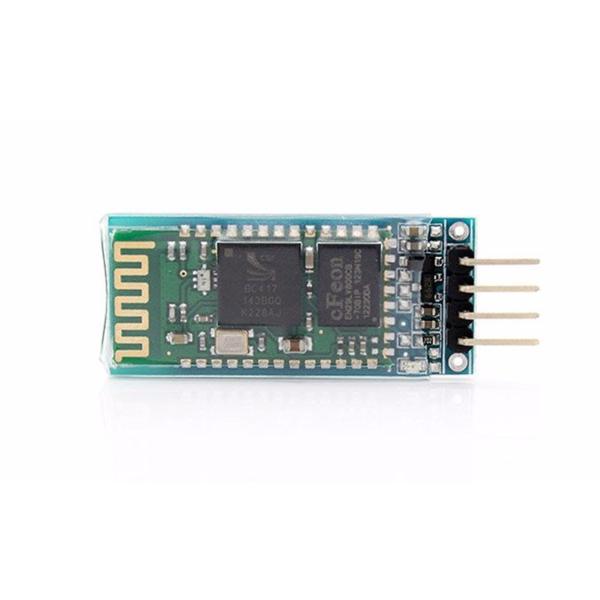
### Sensor Óptico Reflectivo Infrarrojo Cny70

Mediante este sensor obtendremos la reflexión de la superficie pudiendo detectar entre un color u otro (blanco/negro) y así lograr que nuestro dispositivo pueda seguir una línea negra en un fondo blanco o viceversa



### Sensor Ultrasonido Hc-sr04 Distancia

Mediante este sensor obtendremos la distancia de los objetos que se encuentre delante de él, ya que permite ejecutar un ultrasonido y al recibir dicha señal de vuelta luego de haber rebotado en el objeto, calcular la distancia al mismo.



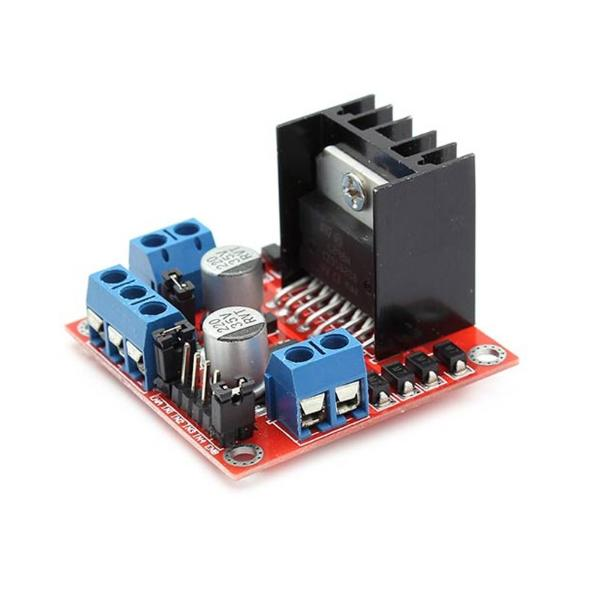
### Modulo Bluetooth Hc06 Uart Ttl Esclavo

Este sensor nos permitirá capturar las señales emitidas con el celular para realizar los cálculos y el funcionamiento necesario.

## Motor Dc 3v A 6v Caja Reductora Rueda Goma Arduino NubbeoActuadores

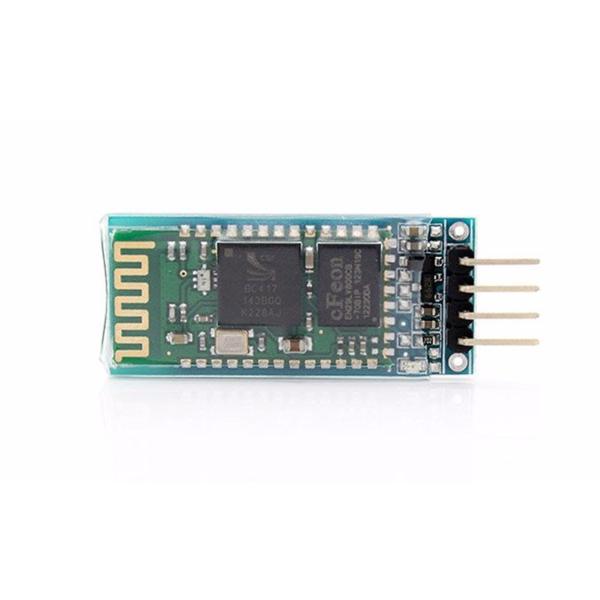
### Motor Dc 3v A 6v Caja Reductora

Dicho motor, junto con la rueda, va a ser el que nuestro dispositivo pueda movilizarse por la superficie.



### Doble Puente H Driver L298n

Este controlador nos permitirá ir variando en la potencia emitida a los motores junto con la dirección de esta.



### Modulo Bluetooth Hc06 Uart Ttl Esclavo

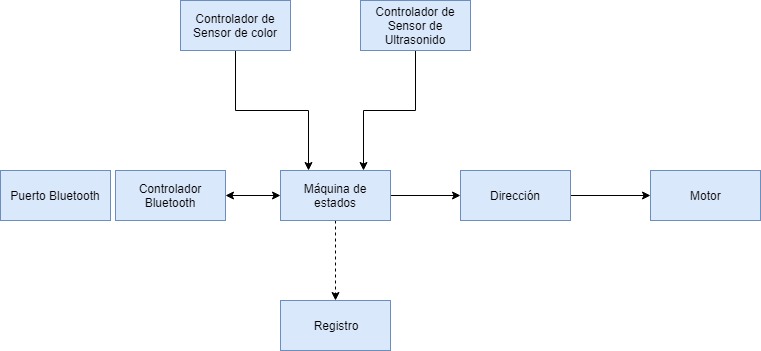
Viéndolo como un actuador, nos permitirá emitir las funciones necesarias al celular según lo que queramos que aparezca en la pantalla.

# Diagramas

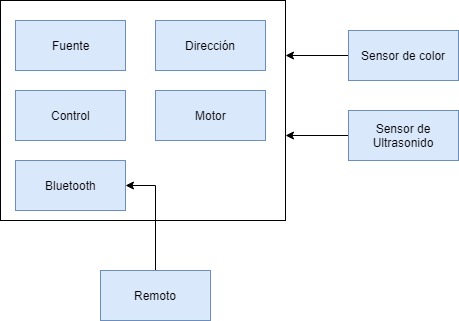
## Diagrama de Estados

## 

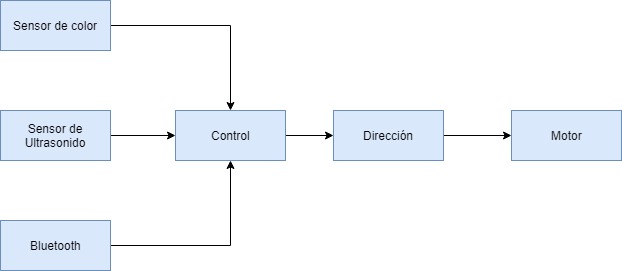
## Diagrama de Software



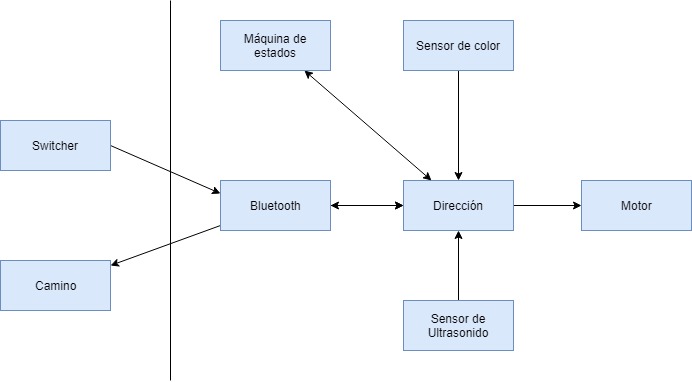
## Diagrama Físico



## Diagrama Funcional



## Diagrama Lógico



# Anexo – Código Arduino