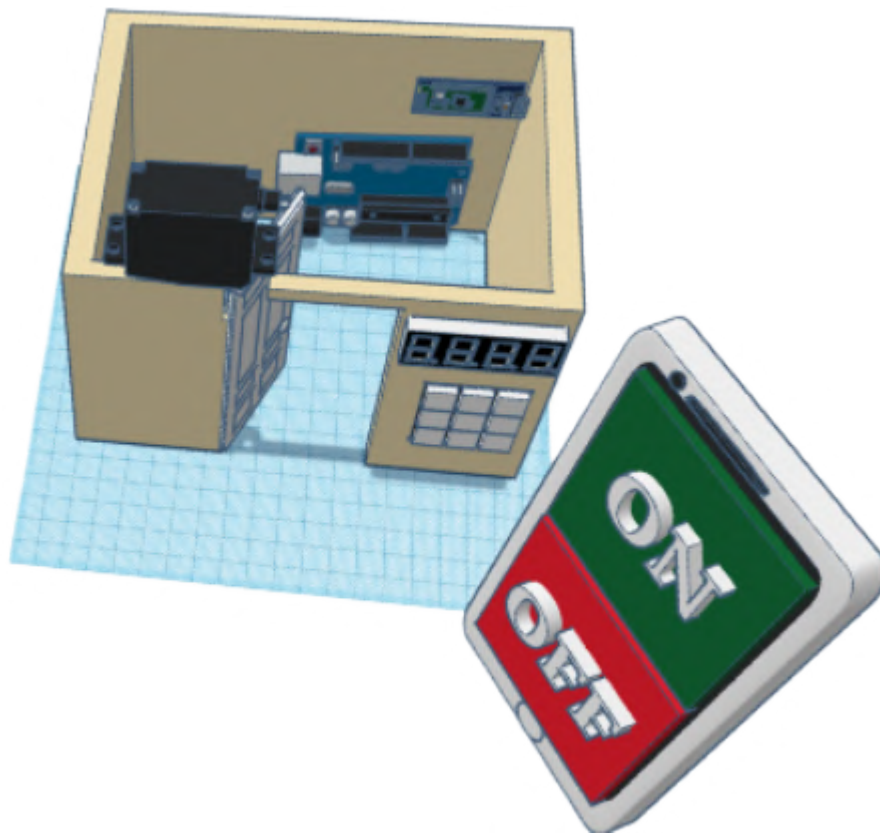




# Institut Puig Castellar

Santa Coloma de Gramenet






## SensorGate Pro

SMX Sistemas Microinformáticos y Redes

David Cajamarca Yugcha  
Iván Martín Bodas  
2SMXD  
2023-2024



Esta obra está sujeta a una licencia de Atribución-NoComercial-CompartirIgual 4.0 Internacional de Creative Common

-  POR: El crédito debe ser dado al creador.
-  NC: Sólo se permiten los usos no comerciales del trabajo
-  SA: Las adaptaciones deben ser compartidas en los mismos términos.

## **Resumen del proyecto:**

El proyecto se centra en automatizar el acceso a las puertas automatizadas mediante bluetooth, pin numérico y sensor utilizando Arduino. El objetivo principal del proyecto es desarrollar un sistema que permita abrir una puerta de forma remota y manual. Para lograrlo, necesitaremos un sensor, teclado numérico, una placa bluetooth y un servomotor para que automatice enviar señales de control desde Arduino.

La metodología de nuestro proyecto se basa en crear una aplicación para poder abrir la puerta mediante el bluetooth, crear los comandos necesarios para que los sensores, el teclado numérico y el bluetooth den señales al servomotor y que se abra la puerta. En el caso de que la contraseña se haya escrito de forma errónea, la aplicación enviará un mensaje al dueño mediante Telegram.

En conclusión, el objetivo de nuestro proyecto se basa en abrir y cerrar una puerta mediante sensor de cercanía, contraseña y a distancia.

## **Palabras clave:**

Automatización, Telegram, comandos, contraseña, puerta remota, Android Studio y Arduino.

## **Abstract:**

The project focuses on automating access to automated doors via Bluetooth, numeric pin, and sensor using Arduino. The project's main objective is to develop a system that allows one to open a door remotely and manually. To achieve this, we will need a sensor, a numeric keypad, a Bluetooth board, and a servomotor to automate sending control signals from Arduino.

The methodology of our project is based on creating an application to open the door via Bluetooth, creating the necessary commands for the sensors, keypad, and Bluetooth to give signals to the servomotor and open the door. In case the password has been mistyped the application will send a message to the owner via Telegram.

In conclusion, the objective of our project is based on opening and closing a door by proximity sensor, password, and remote.

## **Keywords:**

Automation, Telegram, commands, password, remote door, Android Studio, and Arduino

# Índice

<b>1 Introducción</b>	<b>6</b>
1.1 Contexto	6
1.2 Justificación	6
1.3 Objetivos	7
1.3.1 Objetivo general	7
1.3.2 Objetivo específico	7
1.4 Estrategia y planificación del proyecto	8
1.5 Metodología de trabajo	8
1.6 Estudio económico y presupuestario	8
1.6.1 Estudio económico caso real	9
1.6.2 Estudio económico Proyecto	9
<b>2 Descripción del proyecto</b>	<b>12</b>
2.1 Análisis de requisitos	12
2.1.1 Requisitos funcionales	12
2.1.2 Requisitos no funcionales	12
2.2 Tecnologías	13
2.2.1 Comparativa de las tecnologías valoradas	13
2.2.2 Tecnologías Escogidas	14
2.3 Estructura del proyecto	14
2.4 Descripción de los componentes	15
2.4.1 Análisis Detallado de los Materiales	15
2.4.2 Explicación de cada componente	16
Teclado numérico de membrana	16
Especificaciones Teclado numérico	16
Tecnología Bluetooth en Arduino	17
Especificaciones Placa Bluetooth	17
Placa Arduino (ELEGOO UNO R3)	18
Especificaciones placa Arduino	18
Pantalla (LCD Display Azul)	19
Especificaciones placa Arduino	19
Sensor de transductor	20
Especificaciones sensor de transductor	20
Micro Servo	21
Especificaciones Micro servo	21
Protoboard	22
Especificaciones Protoboard	22
Cable de alimentación	23
Especificaciones Cable de alimentación	23
Pila	24
Especificaciones Pila	24
<b>3 Creación de nuestro proyecto</b>	<b>25</b>

3.1 Montaje de la Casa	25
3.1.1 Creación de la casa	25
3.1.2 Configuración de los componentes con Arduino	26
3.2 Código Arduino	29
3.2.1 Explicación código Arduino	31
3.3 Creación de aplicación	33
3.3.1 Método de Instalación de la aplicación	33
3.4 Código de la App	34
3.4.1 Programación Pantalla de Inicio	34
3.4.2 Explicación código pantalla de Inicio	36
3.4.3 Código de Automatización de Apertura y Cierre de Puerta	37
3.4.4 Explicación código de apertura y cierre	39
3.5 Bot de Telegram	40
3.6 Página Web	41
<b>4 Estudio económico proyecto</b>	<b>42</b>
4.1 Análisis del Microentorno	42
4.1.1 Análisis de la competencia	42
4.1.2 Análisis del sector	42
4.1.3 Análisis del DAFO	42
4.2 Idea de Negocio	43
4.3 Estructura organizativa	44
4.3.1 Planificación RR.HH	44
4.3.2 Política personal	45
4.4 Estrategia de Marketing	45
4.5 Plan jurídico fiscal	46
4.6 Plan económico financiero	46
4.6.1 Plan de inversiones iniciales	46
4.6.2 ROI (retorno de la inversión)	47
4.6.3 Previsión de ingresos y gastos	47
<b>5 Problemas a la hora de hacer el proyecto</b>	<b>48</b>
<b>6 Conclusiones</b>	<b>49</b>
6.1 Conclusiones generales del proyecto	49
6.2 Consecución de los objetivos	49
6.3 Valoración de la metodología y planificación	49
6.4 Visión de futuro	50
<b>8 Webgrafía</b>	<b>51</b>
<b>9 Anexos</b>	<b>53</b>

# 1 Introducción

Nuestro proyecto se basa en crear una puerta automatizada que se abra por Bluetooth mediante Android Studio. La puerta automatizada se puede abrir mediante el panel numérico de la aplicación o el físico que hará que se abra la puerta solo si detecta a alguien que está al lado de la puerta. El panel numérico físico se recomienda usar cuando el dispositivo móvil no tenga batería para mejorar la seguridad y privacidad del hogar.

## 1.1 Contexto

Nuestra idea era crear una puerta en Arduino que pudiera abrirse con comandos de voz. Sin embargo, mientras buscábamos información, descubrimos que costaría mucho dinero y sería bastante caro, por lo que nos decidimos a crear una puerta la cual se pudiese abrir de diferentes formas. La forma para acceder a abrir la puerta es con un panel numérico, el cual está situado junto a la puerta. Se puede acceder al panel numérico en físico, que es el que está al lado de la puerta, o de forma digital, con la aplicación desarrollada con Android Studio que va mediante bluetooth. Cuando se haya introducido el pin, si la contraseña es correcta, se activará el sensor y abrirá la puerta; en cambio, si introduces la contraseña errónea, se nos enviará un mensaje de Telegram diciendo que han intentado entrar al hogar (número asignado). Nuestra recomendación es que se use la aplicación en primer caso, pero si no tienes batería en el móvil o en el dispositivo que hayas vinculado la aplicación, tienes la forma física para abrir la puerta con el panel numérico.

## 1.2 Justificación

El proyecto se centra en aumentar la fiabilidad a través de la puerta automatizada mediante un teclado numérico con el fin de verificar la precisión de los accesos, y así evitar que sean no autorizados. Por otra parte, la incorporación de Bluetooth posibilita la apertura de la cerradura desde lejos, esto ofrece comodidad a los usuarios con permiso y aumenta la flexibilidad. Esta propuesta, que se basa en Arduino y componentes electrónicos, ofrece una ocasión para el estudio y la puesta en práctica de los conocimientos en el área de la electrónica y la programación. También, ofrece diversas maneras de ingresar como ser compatible con Bluetooth. El proyecto es necesario para garantizar la eficiencia y la versatilidad de las puertas automatizadas.

## 1.3 Objetivos

El objetivo de este proyecto es diseñar una casa que tenga una puerta automatizada para implementar los métodos de apertura y cierre que se han creado mediante la aplicación desarrollada por bluetooth. El objetivo principal que se pretende es abrir la puerta con la aplicación desarrollada por nosotros que va por bluetooth, la cual necesitamos currirnos de lenguaje de programación para desarrollar la aplicación y que sea funcional. Otro de los objetivos es configurar todo bien en la placa Arduino para que se pueda abrir el panel numérico y salga el pin por pantalla.

### 1.3.1 Objetivo general

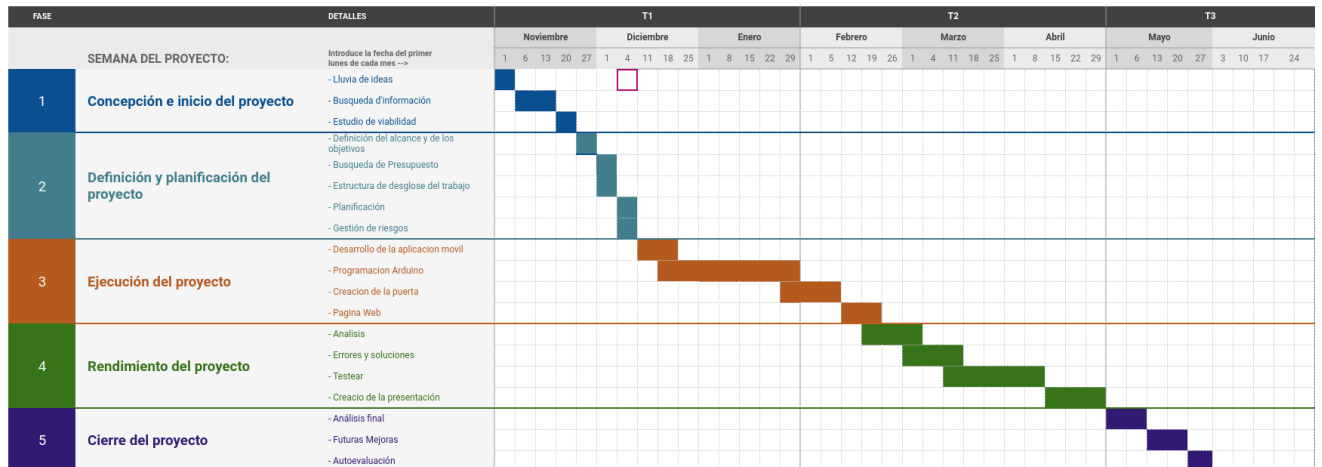
- Configuración Puerta Arduino: El objetivo es aprender a entender el código de Arduino que es parecido a C pero con algunas variables en el código para poder configurar los componentes con la placa Arduino.
- Aprender a hacer una aplicación móvil con comunicación mediante bluetooth: El objetivo es buscar información y nutrirnos de conocimientos para hacer la App con conexión bluetooth
- Desarrollar un bot para Telegram: El objetivo es aprender como se puede enviar un mensaje mediante Telegram cuando escribes la contraseña de forma incorrecta, así mejorando la seguridad del hogar.
- Hacer una página web informativa: El objetivo es hacer una página web con WordPress que ya hemos tocado un poco con Aplicaciones Web de modo que informe sobre dudas de nuestro proyecto.

### 1.3.2 Objetivo específico

- Crear y configurar una puerta automatizada
- Aprender a hacer una aplicación móvil que se comunique con bluetooth
- Desarrollar un bot para Telegram que avise al usuario cuando se escriba incorrectamente la contraseña.
- Poner en práctica nuestros conocimientos con las páginas web utilizando WordPress

## 1.4 Estrategia y planificación del proyecto

La planificación del proyecto se realizará mediante el uso del diagrama de Gantt. Este proporcionará una visión clara de si estamos cumpliendo con los plazos establecidos, evitando así posibles retrasos que hayamos asignado previamente. Gracias al Diagrama de Gantt podremos estructurar mejor la forma de trabajar.



## 1.5 Metodología de trabajo

La metodología que se ha escogido para hacer el proyecto ha sido el modelo de cascada. Este modelo se basa en seguir los pasos sin que puedas avanzar al siguiente sin haber acabado el anterior. Se ha escogido la metodología en cascada, ya que hemos considerado que es la metodología idónea para el proyecto porque si nos fijamos en el diagrama de Gantt el proyecto se centra en diferentes fases para que todo salga adelante y no se puede anteponer la siguiente fase sin haber podido resolver lo anterior.

## 1.6 Estudio económico y presupuestario

El estudio económico que se ha realizado lo hemos dividido en 2 partes: El estudio económico realista y el estudio económico que se ha usado en el proyecto.



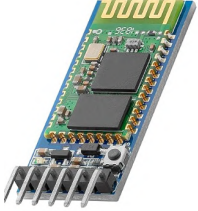



### 1.6.1 Estudio económico caso real

Para hacer el estudio económico se ha tenido en cuenta el conjunto de materiales y el desplazamiento del técnico más la mano de obra.

Producto	Coste
Tecnico (desplazamiento + mano de obra)	300 €
Materiales	300 €
Propinas	30 €
Coste total	630 €

### 1.6.2 Estudio económico Proyecto

Nombre	Imagen	Link de compra	Precio
Teclado numérico de membrana		<a href="#">Link</a>	0,34€
Placa Arduino (ELEGOO UNO R3)		<a href="#">Link</a>	19,99€
Placa Bluetooth		<a href="#">Link</a>	9,99€
Pantalla (LCD Display Azul)		<a href="#">Link</a>	4,40€

<p>Micro Servo</p>		<p><a href="#">Link</a></p>	<p>3,48€</p>
<p>Cable M/M H/M H/H</p>		<p><a href="#">Link</a></p>	<p>1,00€</p>
<p>Protoboard</p>		<p><a href="#">Link</a></p>	<p>1,00€</p>
<p>Sensor</p>		<p><a href="#">Link</a></p>	<p>9,00€</p>
<p>Cable de alimentación</p>		<p><a href="#">Link</a></p>	<p>0,83€</p>
<p>Batería</p>		<p><a href="#">Link</a></p>	<p>5,98€</p>
<p>Madera balsa</p>		<p><a href="#">Link</a></p>	<p>6,30€</p>
<p>Total</p>			<p>62,30€</p>

## 2 Descripción del proyecto

El proyecto se basa en desarrollar una aplicación con Android Studio para que pueda abrir la puerta automatizada mediante bluetooth y otra manera de apertura es con el panel numérico físico que está situado al lado de la puerta. En caso de escribir mal la contraseña el bot de Telegram enviará un mensaje avisando de que se ha introducido una contraseña errónea.

### 2.1 Análisis de requisitos

Para sacar el proyecto adelante ha de funcionar muchas cosas que cada una tarda mucho tiempo en ser desarrollada.

- Configurar la placa Arduino para que todos los componentes se vean entre sí y puedan trabajar en conjunto. Para que el panel numérico esté vinculado y salga la contraseña por pantalla.
- Desarrollar la aplicación con Android Studio para que se pueda abrir la puerta automatizada mediante bluetooth y pueda enviar un mensaje al bot de Telegram en caso de que la contraseña se haya introducido de forma errónea.
- La página web ha de funcionar para que se pueda buscar información al respecto de SensorGate Pro.

#### 2.1.1 Requisitos funcionales

- Conectividad de la App mediante bluetooth con la placa de Arduino para abrir y cerrar la puerta automatizada.
- El panel numérico puede abrir la puerta automatizada con el panel numérico físico y con la aplicación móvil mediante bluetooth.
- Integración de la aplicación móvil con Telegram para enviar alertas

#### 2.1.2 Requisitos no funcionales

La página web tiene un carácter informativo sobre el proyecto, explica un poco nuestra empresa cómo surgió la idea, el procedimiento del montaje e información adicional al respecto de SensorGate Pro.

## 2.2 Tecnologías

### 2.2.1 Comparativa de las tecnologías valoradas



#### App Inventor vs Android Studio



App Inventor es una tecnología con interfaz gráfica dedicada a principiantes en programación, ya que te facilita programar mediante bloques arrastrando y soltando aplicaciones para Android.

Android Studio es la tecnología IDE oficial para Android, es un entorno de desarrollo el cual tienes que escribir el código a mano con el lenguaje de Kotlin + Java, ya que son los dos lenguajes oficiales para Android Studio.

Si estás introducido en el tema de los lenguajes de programación y quieres desarrollar una aplicación compleja recomiendo el uso de Android Studio, ya que es mucho más completo que App Inventor en tema de limitaciones.



#### Google Sites vs WordPress

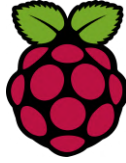


Google Sites es un servicio de Google para crear páginas web. Google Sites permite crear una página web sin tener que programar y usar código HTML, haciendo clics y te puedes hacer la página web a tu gusto.

WordPress es un sistema de gestión de contenido para administrar y crear páginas web. Se suele utilizar para crear páginas web en tiendas online, ya que hay mucha variedad de plugins que son muy útiles para hacer tiendas online como por ejemplo WooCommerce.



Arduino vs Raspberry Pi



Arduino es una compañía basada en Hardware libre que se dedica a hacer placas base para que luego los clientes puedan crear un circuito electrónico basado en el diferente tipo de placa, porque hay diseñadas muchos tipos de placa cada una para una función diferente. El lenguaje que se usa de programación en Arduino es C o C + +.

Raspberry Pi es una placa de un microordenador que hace las funciones básicas de un ordenador buscar en internet y más la cual se puede conectar a un monitor y un teclado porque tiene entradas USB y eso es una de sus puntos fuertes que es que tiene varias entradas y salidas.

## 2.2.2 Tecnologías Escogidas

Hemos elegido usar Android Studio, ya que es un reto aprender un lenguaje de programación y nos resultaba muy curioso aprender a programar y entender el lenguaje de Java y Kotlin que son los oficiales para Android Studio. App Inventor era trabajar mediante bloques y nos resultaba demasiado sencillo para presentarlo en un proyecto final.

Se ha elegido WordPress por la misma razón que nos hemos decantado por Android Studio que ha sido la complejidad. El motivo de la elección de WordPress es por la variedad que tiene en comparación con Google Sites y las diversas opciones que tienes para hacer y administrar la página web a tu gusto.

Se ha elegido Arduino porque lo que queríamos hacer es aprender C ++, ya que es más difícil de entender que Python que es el lenguaje de Raspberry Pi y en términos de componentes era mejor usar Arduino para la implementación.

## 2.3 Estructura del proyecto

La estructura del proyecto se basa en dos pilares fundamentales que es la placa de Arduino y por otro lado la aplicación móvil.

La estructura del proyecto es hacer una casa que tenga adentro los componentes conectados para que se comuniquen entre ellos y en los apartados posteriores se explicará cómo están conectados todos los componentes.

Hay dos métodos de apertura:

El Modo de apertura con el panel numérico físico sirve para introducir una contraseña que se la envía a Arduino para ver si es correcta. La placa Arduino mirara si le ha llegado la señal del panel numérico y si le llega la envía a la pantalla LCD para que salga por pantalla la contraseña y pueda verificar si la contraseña se ha escrito de manera correcta. Al poner la contraseña si hay una presencia que sienta el sensor enviará otra señal al Arduino que la recibirá el servomotor y se encargará de abrir la puerta. Todo esto es gracias al cable de alimentación y la pila que es lo que dará energía al proyecto.

El otro método de apertura es gracias a la aplicación que va mediante bluetooth y es muy parecido. En primer lugar si escribes el pin saldrá por pantalla y se podrá ver si es correcta o no dándole a bolton de acceder, si es correcta pasará a otra pantalla para poder vincular el bluetooth del móvil con la placa bluetooth que está conectada al Arduino. Al vincular la placa con el dispositivo móvil tendrás el control de la puerta automatizada mediante señales bluetooth. En el caso de que la contraseña introducida sea errónea se enviará una señal al bot de telegram para avisar que se ha introducido la contraseña errónea y que tenga cuidado en su hogar.

## 2.4 Descripción de los componentes

### 2.4.1 Análisis Detallado de los Materiales

**Teclado numérico de membrana:** Utilizaremos un teclado de membrana para conectarlo a la Placa Arduino, la cual estará programada para que nos permitan abrir la puerta mediante contraseña.

**Placa Arduino:** La Placa Arduino es lo más importante de los materiales que hemos comprado, ya que toda la programación y componentes estará dentro de este.

**Placa Bluetooth:** Como dice su nombre, la placa bluetooth la utilizaremos para abrir remotamente la puerta, mediante un dispositivo móvil, que tenga instalada la aplicación creada por nosotros.

**Pantalla (LCD):** Es una Pantalla de dimensión de 80x36x12 que nos permitirá ver los dígitos que coloquemos con el teclado numérico, está conectado al Arduino y programado para que detecte el teclado numérico.

**Micro Servo:** El servo motor es un minimotor que su función en nuestro proyecto es desbloquear la puerta.

**Cable M/M H/M H/H:** Estos cables nos sirven para conectar los componentes a la placa Arduino.

**Protoboard:** Es una placa de pruebas que permite montar y desmontar circuitos electrónicos de forma muy fácil.

**Sensor:** El sensor nos servirá para que la puerta se abra y se cierre al momento de abrirlo por el teléfono o por el panel numérico, este sensor está conectado en la placa Arduino.

**Cable de alimentación:** El cable de alimentación nos servirá para alimentar la placa de Arduino con la batería.

**Pila:** La pila nos servirá para que no esté conectado directamente al ordenador y podamos transportarlo con más facilidad y con energía.

**Madera balsa:** La madera balsa es un tipo de material muy ligero y suave, el cual se usa principalmente para hacer manualidades y proyectos.

## 2.4.2 Explicación de cada componente

### Teclado numérico de membrana

El teclado numérico de membrana es un aparato electrónico que se utiliza para enviar señales. Las señales que mandan se producen gracias a las membranas que tiene en la parte superior, las cuales abajo están los circuitos electrónicos para que funcionen correctamente y mande las señales al dispositivo conectado en nuestro caso a la placa de Arduino.



### Especificaciones Teclado numérico

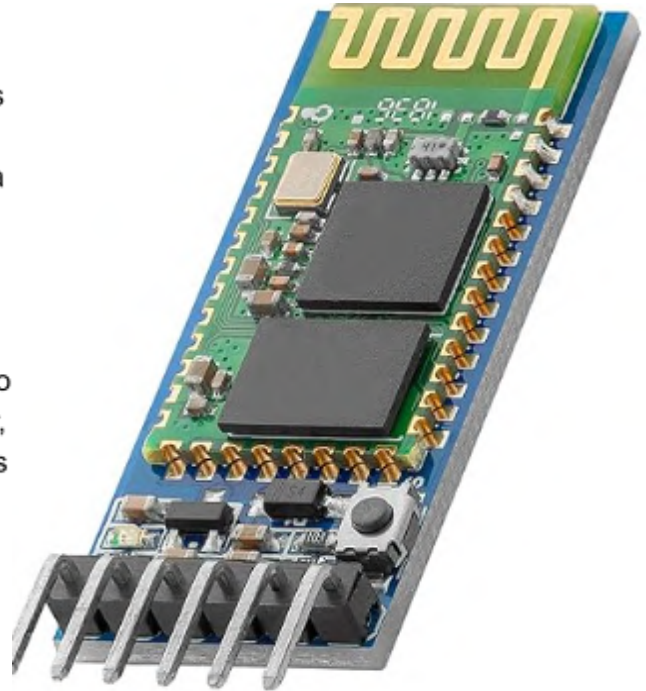
- Resistencia de contacto: 10 a 500 ohmios.
- Resistencia de aislamiento: 100 megaohmios a 100 voltios.
- Fuerza operativa de la tecla: 150 a 200 Newtons.
- Tiempo de rebote: 1 milisegundo.
- Vida útil: 100 millones de veces.
- Temperatura de funcionamiento: -40 a +80 grados Celsius.



## Tecnología Bluetooth en Arduino

La tecnología bluetooth da señales de forma inalámbrica hacia los dispositivos conectados al Arduino para desbloquear la puerta de forma remota. Gracias a la automatización, la puerta se puede abrir y cerrar con la tecnología bluetooth para dar facilidades a la hora del acceso.

Hemos decidido decantarnos por este módulo bluetooth por varias razones. En primer lugar, nos interesó la opción que tiene esta placa es que es transceptor, entonces actúa como transmisor o receptor y puede conectarse a varios aparatos electrónicos gracias a los cables que se posicionan en los 6 pines que se ven en la imagen.

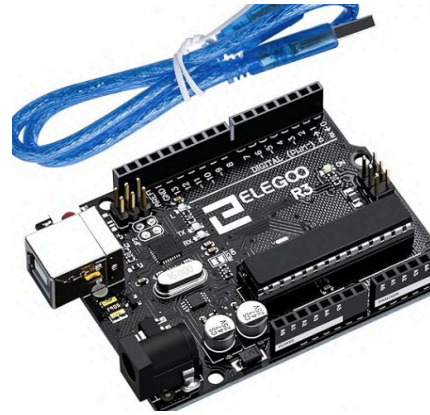


### Especificaciones Placa Bluetooth

- Alcance de hasta 10 metros: Utilizando Bluetooth de clase 2, este módulo tiene un alcance de hasta 10 metros.
- Modo emisor y receptor: Puede utilizarse tanto como emisor como receptor.
- LED de estado: El módulo tiene un LED que muestra el estado:
  - Parpadeo rápido: Listo para la conexión.
  - Parpadeo lento: Conexión establecida.
  - Parpadeo a intervalos de 2 segundos: Modo AT (puede programarse a través del IDE de Arduino)

## Placa Arduino (ELEGOO UNO R3)

Hemos elegido esta placa de Arduino porque es la más completa para nuestro proyecto porque encaja con las necesidades que tenemos para hacer nuestro proyecto. La placa es una ELEGOO UNO R3. Nos hemos decantado por este modelo, concretamente debido a que tiene una placa con microcontrolador y un software que nos facilita programar y desarrollar el proyecto de nuestra puerta automatizada. Esta placa está enfocada para la automatización de proyectos.

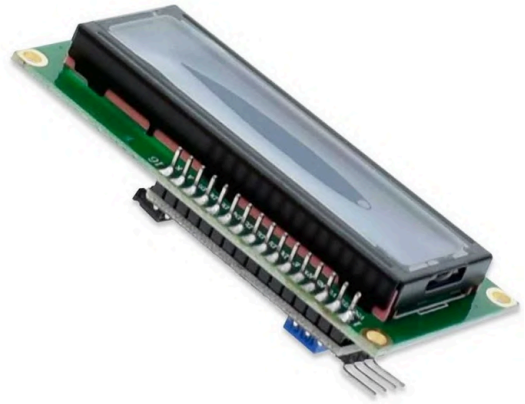


### Especificaciones placa Arduino

- Microcontrolador: mega
- Voltaje de entrada: 7-12V
- 5V de corriente eléctrica: 500MA
- Corriente eléctrica 3.3V: 50MA
- 14 pines de I/O digitales (6 salidas PWM)
- 8 entradas analógicas
- Memoria Flash de 32k
- 16Mhz Clock Speed

## Pantalla (LCD Display Azul)

La pantalla LCD Display Azul es una pantalla de cristal líquido con una interfaz I2C. En nuestro proyecto, usaremos esta pantalla para mostrar la contraseña numérica que introducimos con el teclado numérico de membrana.



### Especificaciones placa Arduino

- Características de la pantalla:
  - Pantalla LCD, 16 caracteres y 2 líneas.
  - Alimentación 5V.
  - Controlador compatible HD44780.
  - Fondo azul retroiluminado, letras en blanco.
  - Incluye un pin de contraste ajustable.
  - Dimensiones (mm): 80 x 36 x 12.
- Característica del adaptador:
  - Basado en el chip de PCF8574T.
  - Alimentación: 5V.
  - Soporta Protocolo I2C.
  - Un potenciómetro para regular el contraste.
  - La dirección I2C predeterminado es 0x20.
  - Se puede cambiar a cualquiera entre 0x20 y 0x27 según tengas soldados o no los pines A0 A1 A2.
  - Dimensiones: 19 x 42 mm.
  - Versión Bulk (envío sin embalaje, se envía en un sobre y una bolsa de plástico).

## Sensor de transductor

Los sensores que utilizamos son de transductor, los cuales su función es recibir y responder a las señales recibidas por el Arduino. La cual hará que la puerta detecte si hay alguna presencia, y se nota la presencia se abrirá si no se quedará cerrada.



## Especificaciones sensor de transductor

- Voltaje de trabajo: 5V (DC)
- Corriente estática: menos de 2mA.
- Señal de salida: señal de frecuencia eléctrica, nivel alto 5V, nivel bajo 0V.
- Ángulo del sensor: no más de 15 grados.
- Distancia de detección: 2cm-450cm.
- Alta precisión: hasta 0,3 cm.
- Señal de activación de entrada: 10us impulso TTL
- Señal de eco: señal de salida TTL PWL

## Micro Servo

El micro servo es un mini motor que nos servirá para que la puerta se abra y se cierre automáticamente. Este motor se usa normalmente en proyectos de Arduino porque es muy útil para hacer según qué movimientos. El uso que le daremos será para abrir y cerrar la puerta con instrucciones, variables...



### Especificaciones Micro servo

- Voltaje de trabajo: 5V (DC)
- Corriente estática: menos de 2mA.
- Señal de salida: señal de frecuencia eléctrica, nivel alto 5V, nivel bajo 0V.
- Ángulo del sensor: no más de 15 grados.
- Distancia de detección: 2cm-450cm.
- Alta precisión: hasta 0,3 cm.
- Señal de activación de entrada: 10us impulso TTL
- Señal de eco: señal de salida TTL PWL
- Modo de conexión: VCC / trig (T) / Echo (R) / GND

## Protoboard

Un protoboard es una placa de pruebas con agujeros conectados entre ellos. La placa puede conectar los componentes electrónicos para crear prototipos y si algo falla poder cambiarlo sin complicaciones como dice el nombre es una placa para realizar pruebas. Nosotros los utilizamos para conectar nuestros componentes entre sí.



### Especificaciones Protoboard

- Placa de pruebas sin soldadura de 170 / 400 / 830 puntos de amarre, prototipo de placa de pruebas Dupont Universal PCB Jumper, placa de prueba Dupont de 12.
- Compatible con Raspberry Pi (RPI) y Arduino, diodo LED, terminal de tornillo, cable DuPont, conector, resistencia, IC, DIP, sensores, transistor y otros componentes o dispositivos de energía eléctrica.

## Cable de alimentación

El cable de alimentación es un cable el cual su función principal es alimentar cualquier circuito electrónico. Este cable en concreto da 9 V de salida y es compatible con nuestra placa ELEGOO UNO R3.

El cable de alimentación tiene una parte macho que es la que se conecta con la placa de Arduino y la otra parte es donde se pone la batería.



## Especificaciones Cable de alimentación

- Compatibilidad: Apto para Arduino Uno R3.
- Longitud del cable: 15 cm.
- Conector 1: Tipo T, ideal para baterías de 9V.
- Conector 2: DC 5,5x2,1mm, estándar para muchos dispositivos electrónicos.
- Voltaje: Adaptación para DC 9V.
- Materiales: Fabricación de alta calidad para mayor durabilidad.

## Pila

La pila alcalina de 9 V es una parte fundamental en el proyecto porque se encarga de alimentar a toda placa de Arduino. Hemos elegido este modelo en concreto porque hemos estado buscando modelos calidad precio y era el que más nos convencía de la gama de 9 V.



## Especificaciones Pila

- Voltaje: 9V
- Tipo: Alcalina
- Código de modelo: 6LR61/MN 1604
- Duración: 10 años



## 3 Creación de nuestro proyecto

### 3.1 Montaje de la Casa

#### 3.1.1 Creación de la casa

Para la creación de la casa se han usado 3 tablones de madera contrachapada A3 30X40 y con un lápiz se ha hecho el boceto que tenía que tener la puerta y los diferentes componentes para no tener problemas de espacio a la hora de recortar la madera.

Al cortar la madera con una sierra no salió perfecto, ya que no somos profesionales y era la primera vez que trabajábamos con ello, entonces tuvimos que coger papel de lija para quitar los imperfectos que tenía la madera. Después de lijar venía la parte difícil porque no sabíamos cómo hacer los agujeros para el sensor hasta que nos dimos cuenta de que lo único que teníamos para hacer los agujeros era un taladro y nos pusimos manos a la obra.

Al ya tener las paredes de la casa recortadas, pasamos a pegar las paredes con la pistola de silicona para vidrios que era lo único que teníamos en casa que haga la función de pegar.

Imagen del resultado de las paredes



Imagen pegando las paredes



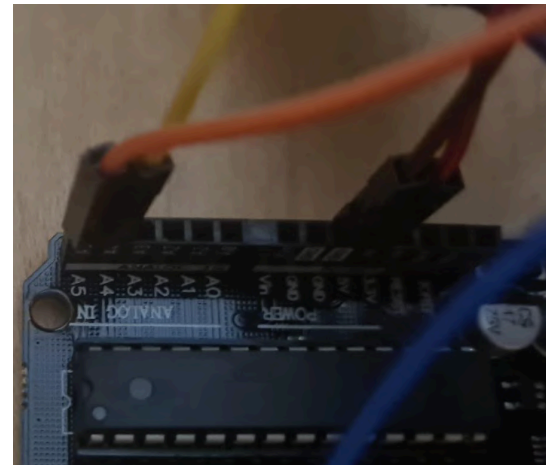
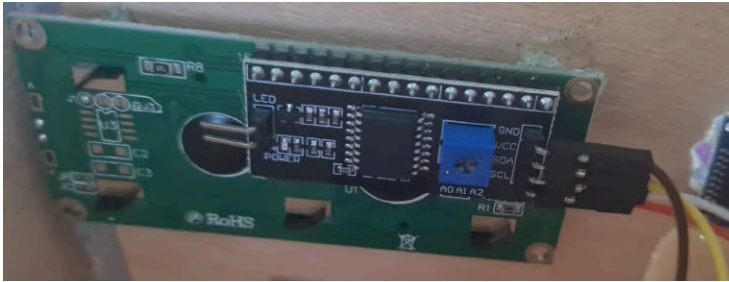
### 3.1.2 Configuración de los componentes con Arduino

Para configurar los componentes con la placa base de Arduino primero hay que ver cómo se conectan entre ellos.

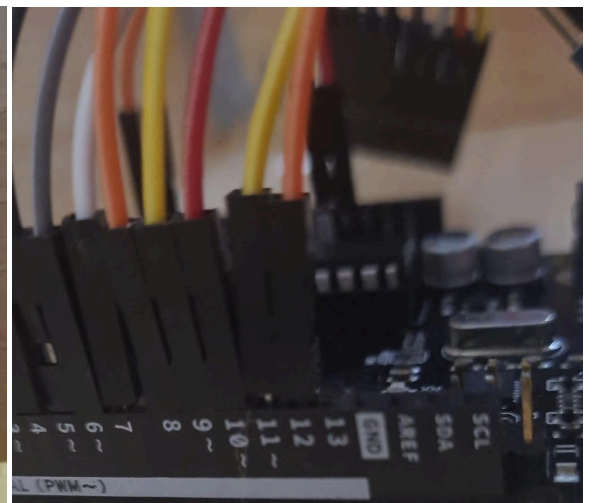
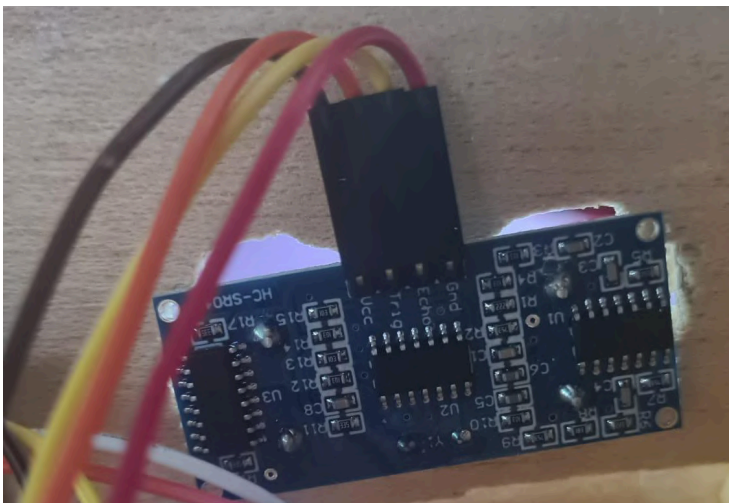
En primer lugar decidimos poner la placa base y ya ir conectando los componentes a partir de ahí.

Primero de todo se ha conectado un cable en GND y 5V en el protoboard cada uno en diferentes filas para que se pueda conectar los diferentes componentes a los voltios que necesita y ahora empezamos con el montaje de los componentes.

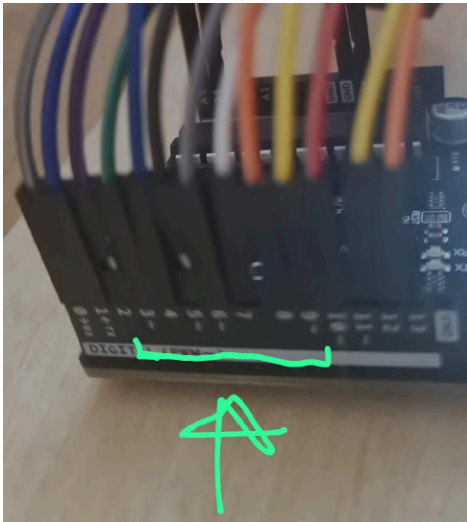
La pantalla LCD tiene 4 pines y se van a explicar de arriba hacia abajo los diferentes pines. El pin más alto en la imagen que es el pin GND está conectado al GND del protoboard, el segundo pin es el VCC y se ha conectado en la protoboard en el apartado de 5V. El siguiente pin es el SDA y se ha conectado en el pin A4 y por último el pin que queda que es el SCL está conectado al A5 de la placa base.



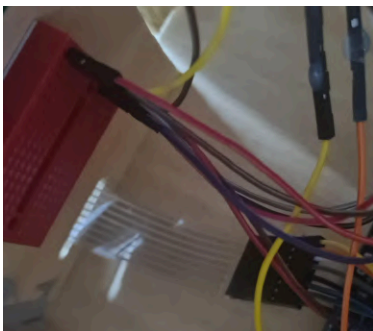
La conexión del sensor también parte de 4 pines el GND va conectado al protoboard en el sector de GND. El pin echo está conectado en el pin 11 de la placa de Arduino. El pin Trig se conecta al lado del pin echo ya que está en la posición 12 y por último el VCC se sitúa en el protoboard de 5V.



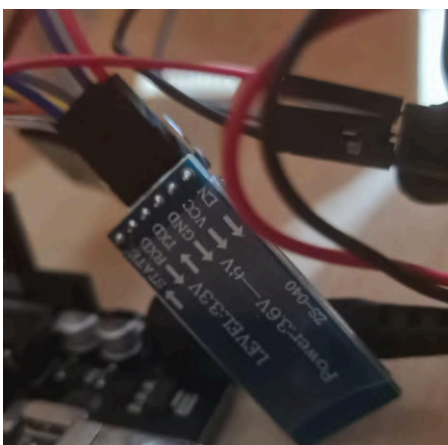
El panel numérico tiene una distribución de pines sencilla y se ha puesto en la placa de Arduino del pin 2 al pin 9 para que no haya ningún problema a la hora de tocar el panel numérico físico y no haya alguno descolgado por colocarlos de manera desordenada.



En el interior de la casa se puede ver cómo está conectado todos los pines al panel numérico sin que haya ninguna interferencia



La configuración bluetooth también dispone de 4 pines los cuales el VCC está conectado a los 5V de la protoboard, el GND también va a la protoboard, pero al apartado de GND, el pin TXD que no se había visto anteriormente va conectado al 0 RX de la placa base de Arduino y el RXD va conectado al 1TX de Arduino.



Por último nos queda la configuración del servomotor que es lo que hará que la puerta se abra o no. El cable de color naranja va conectado al pin 10 de la placa base de Arduino, el cable rojo va al GND del protoboard y el cable marrón va conectado a los 5V del protoboard.



El cable de alimentación va conectado a la placa de Arduino para que se pueda conectar la pila no se ha puesto porque no queremos que se gaste la pila. Va conectado donde se ve la flecha en la imagen.



Esta ha sido la configuración que se ha tenido que hacer para hacer funcionar el proyecto.

## 3.2 Código Arduino

```

#include <Servo.h>
#include <Wire.h>
#include <Keypad.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);

//Esto se utiliza para configurar el servo motor y la matriz del panel numérico
Servo servolocked;
const byte filas = 4;
const byte cols = 4;
byte Pines_Filas[] = {5, 4, 3, 2};
byte Pines_Cols[] = {9, 8, 7, 6};

//Esto sirve para detectar qué tecla se ha tocado en el panel numérico
char Teclas[filas][cols] =
{
  {'1', '4', '7', '+'},
  {'2', '5', '8', '0'},
  {'3', '6', '9', '#'},
  {'A', 'B', 'C', 'D'},
};

//Aquí se indica cual es la clave para acceder
char codigoSecreto[4] = {'B', '0', 'C', '4'};

//Aquí se nombre 3 variables
int posicion = 0;
int cursor = 5;
int clave = 0;

// Aquí es donde estan conectados los pines del sensor
const int trigPin = 12;
const int echoPin = 11;

//Esto se utiliza para interactuar con el panel numérico
Keypad Teclado1 = Keypad(makeKeymap(Teclas), Pines_Filas, Pines_Cols, filas, cols);

//Este código está configurando la comunicación serial, inicializando y configurando un
LCD,
//y conectando un servo a un pin digital.
void setup() {
  Serial.begin(9600);
  lcd.init();
  lcd.backlight();
  lcd.print("CODIGO:");
  servolocked.attach(10);
  pinMode(trigPin, OUTPUT); // Configura el pin Trig como salida
  pinMode(echoPin, INPUT); // Configura el pin Echo como entrada
}

void loop() {
  char pulsacion = Teclado1.getKey(); //Esta linea lee la tecla pulsada
  if (pulsacion != 0) { //Esta line detecta si se ha pulsado alguna tecla
    // Ejecuta la variable que indica que no se a
  }
}

```

```

//ingresado la clave correcta
if (clave == 0) {
// indica que no aparezca #,*
if (pulsacion != '#' && pulsacion != '*') {
lcd.print(pulsacion);
cursor++;

if (pulsacion == codigoSecreto [posicion]) {
posicion++;
}

// Lee si los cuatro dígitos coinciden con la clave secreta
if (posicion == 4) {
lcd.clear();
lcd.print(" Clave correcta");

clave = 1;
//Esta parte del código se utiliza para medir la distancia de un objeto
digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    long duration = pulseIn(echoPin, HIGH);
    long distance = (duration/2) / 29.1;

// Si el objeto está a menos de 10 cm
    if (distance < 10) {
        servolocked.write(180);

    }

//Una vez se abra la puerta se cerrará en 10 segundos
delay(10000);
servolocked.write(0);
clave = 1;
}

}

//Si la clave es incorrecta aparecerá el siguiente mensaje
if (cursor>8) {
posicion = 0;
lcd.print(" ");
if (clave ==0) {
lcd.clear();
lcd.print ("Clave Incorrecta");
}}}

```

```

//Este código borrar la clave de la pantalla
if (pulsacion == '*') {
  posicion = 0;
  cursor = 5;
  clave = 0;
  lcd.clear();
  servolocked.write(0);
  lcd.print(" ");
}

//Aquí te detecta si envía el mensaje bluetooth para abrir y cerrar la puerta
if (Serial.available()) {
  char command = Serial.read();
  switch (command) {
    case 'C':
      servolocked.write(0);
      break;
    case 'O':
      servolocked.write(180);
      break;
  }
}
}

```

### 3.2.1 Explicación código Arduino

- Lo primero que se puede ver en el código son las librerías Servo, Keypad que es el panel numérico y del LCD estas librerías son necesarias porque son herramientas que nos ayudaran con la programación del Arduino.
- Después nombramos el Servolock que es la variable del servo, el nombre de servolocked se ha de poner si lo nombramos en el código, luego hemos puesto cuántas filas y columnas tiene nuestro teclado que sería de 4 filas y 4 columnas, también definimos en que pines están conectados las columnas y filas de nuestro panel numérico.
- En el siguiente cuadro de código nombramos las teclas para que cuando hagamos clic en el panel numérico, mande la información al Arduino.
- En la línea de codigosecreto indicamos cuál sería la clave para abrir la puerta, debajo de este código nombramos 3 variables que utilizaremos a menudo también indicamos en que pines va el sensor.
- En la siguiente línea de código nombramos Teclado1 a la variable Keypad y unimos los pines con las filas y columnas para que cuando presionemos una tecla salga el carácter correcto.

- La parte del **void setup** está configurado para que se ejecute cuando se encienda o se reinicie la parte de **serial.begin(9600)** este se encarga de leer los datos que envíe el panel numérico, el `lcd.init` inicia la pantalla LCD y el `lcd.backlight` inicia el contraste de la pantalla, el `lcd.print("CODIGO")` ahora aparece en la pantalla LCD la Palabra código y la siguiente línea define el pin del servomotor por último las dos últimas líneas indica la entrada y salida del sensor.
- Parte de **void loop** las dos primeras líneas que se ha puesto en void loop se utilizan para leer y verificar las teclas aplastadas en el panel numérico, la siguiente línea indica la variable que indica que la clave es incorrecta, las 5 líneas de código siguiente hacen la función para que no aparezca en pantalla las teclas # y \* pero que sí hagan su función que explicaré más adelante.
- La siguiente sección del código de **void loop**, lee si los cuatro dígitos del código son correcto si son correctos enviará un mensaje "Clave correcta", activando el sensor que si detecta que hay alguien a menos de 10 cm gusto cuando ponen la clave activará el servo motor que dará un giro de 180 grados y después de 10 segundos se volverá a su lugar cerrando la puerta, si la clave es incorrecta saldrá un mensaje diciendo que la "clave es incorrecta" y no hará nada.
- La penúltima parte del código hace que el botón \* borre lo que esté puesto en la pantalla.
- La última parte del código sirve para la aplicación a qui detecta que si a enviado el mensaje "C" que seria cerrar la puerta lo el servo motor se moverá a 0 grados y si el mensaje es O abrirá la puerta y el servo motor movería 180 grados.

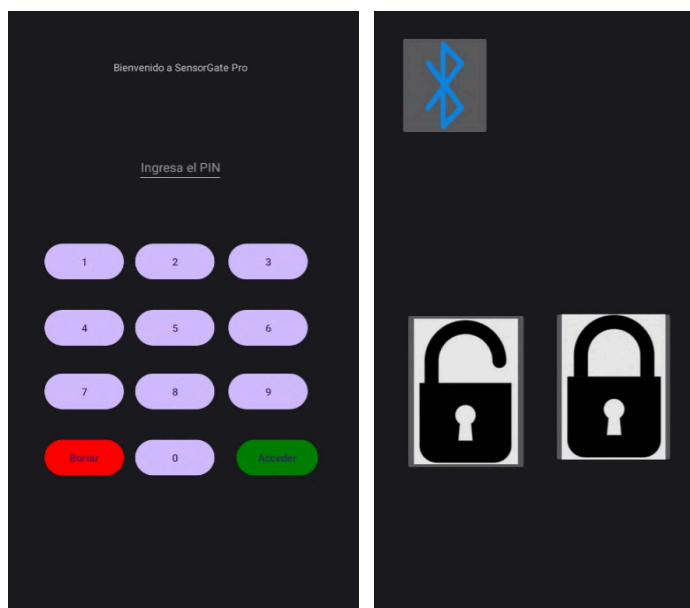


### 3.3 Creación de aplicación

Para crear la aplicación se utiliza la aplicación Android Studios. Esta aplicación permite probar y escribir código además de poder emular un teléfono Android para probar las aplicaciones que se van desarrollando.



En nuestra aplicación, se puede observar dos apartados: En el primer apartado nos mostrará un panel de código que tendremos que colocar el código de número correcto si se coloca un código erróneo enviará un mensaje al propietario de la casa. En el segundo apartado hay dos botones uno que es un candado cerrado que cierra la puerta y otro de un candado abierto que abre. La aplicación irá mediante bluetooth y estará vinculada a la placa de Arduino del hogar.



#### 3.3.1 Método de Instalación de la aplicación

Para poder instalar la aplicación habrá que contactarnos antes para recibir el servicio de la puerta automatizada y cuando se vaya a montar la puerta instalaremos la aplicación en el dispositivo que nos diga el cliente con la contraseña a elección propia.

## 3.4 Código de la App

### 3.4.1 Programación Pantalla de Inicio

```

package com.example.sensor

import android.content.Intent
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import kotlinx.coroutines.CoroutineScope
import kotlinx.coroutines.Dispatchers
import kotlinx.coroutines.launch
import java.net.HttpURLConnection
import java.net.URL

class MainActivity : AppCompatActivity() {

    //Aquí se declara la variable editText y Button
    private lateinit var editText: EditText
    private lateinit var borrar: Button

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        //Aquí se indica que editText es igual a la id editTextTextPassword
        editText = findViewById(R.id.editTextTextPassword)

        //En este recuadro se envía un mensaje a un chat de telegram
        fun sendTelegramMessage(chatId: String, text: String, botToken: String) {
            CoroutineScope(Dispatchers.IO).launch {
                val urlString =
                    "https://api.telegram.org/bot$botToken/sendMessage?chat_id=$chatId&text=$text"
                val url = URL(urlString)
                val conn = url.openConnection() as HttpURLConnection
                conn.requestMethod = "GET"
                if (conn.responseCode != 200) {
                    println("Failed to send message. Response Code:
                    ${conn.responseCode}")
                }
            }
        }
    }
}

```

```

//Aquí se indica que acceder es igual a la id acceder
val acceder: Button = findViewById(R.id.acceder)

//En este recuadro se indica que si la contraseña
// es igual a 1234 se abrirá la siguiente pantalla
// de lo contrario se enviará un mensaje a un chat de telegram-
acceder.setOnClickListener {
    val password = editText.text.toString()
    if (password == "1234") {
        val intent: Intent = Intent(this, Abrir_Cerrar:: class.java)
        startActivity(intent)
    } else (sendTelegramMessage("7150768725", "Alguien quiere entrar a tu
casa", "7050245334:AAHY0ws2Dgvgmg7jnFD6-4HwjBHLjLC0IDA"))
}

//En este recuadro se declaran los botones del 0 al 9
val buttons = arrayOf(

    findViewById<Button>(R.id.button_0),
    findViewById<Button>(R.id.button_1),
    findViewById<Button>(R.id.button_2),
    findViewById<Button>(R.id.button_3),
    findViewById<Button>(R.id.button_4),
    findViewById<Button>(R.id.button_5),
    findViewById<Button>(R.id.button_6),
    findViewById<Button>(R.id.button_7),
    findViewById<Button>(R.id.button_8),
    findViewById<Button>(R.id.button_9))

//Aquí se indica que borrar es igual a la id borrar
borrar = findViewById(R.id.borrar)

//En este recuadro se indica que al dar click en el botón borrar se limpiara el editText
borrar.setOnClickListener {
    editText.text.clear()

//Este recuadro sirve para que cuando
//hagan click en los números aparezcan en el editText
buttons.forEachIndexed { index, button ->
    val number = (index + 0).toString()
    button.setOnClickListener {
        editText.append(number)
    }
}
}
}
}

```

### 3.4.2 Explicación código pantalla de Inicio

- La primera parte del código sería los import que son librerías que se utilizan para diferentes funciones como los botones, cuadros de texto, etc.
- Lo segundo sería dos líneas de código que nombramos dos variables el edit text y el botón
- La siguiente parte del código indica que editText sería igual a id editTextTexPassword la otra parte del código es la configuración de Telegram trata de un enlace que recopila información del botToken que es como la id del bot de Telegram, chatId que sería el id del chat donde enviará el mensaje y texto que es el texto que vamos a escribir más adelante y si falla al enviar el mensaje aparecerá un mensaje de error.
- Después nombrando otro botón con el nombre acceder que lo programaremos para que si lo presionan y la contraseña es correcta salte a la siguiente pantalla de lo contrario, enviará el mensaje de Telegram, al chat **7150768725** con el bot **peligro** que se ha creado en la aplicación de Telegram.
- Los siguientes códigos indican que el botón que se llama borrar tiene el id.borrar y lo configuramos para que elimine el texto.
- El último código sirve para que los números empiecen contando desde 0 y que los números salgan al editText.

## 3.4.3 Código de Automatización de Apertura y Cierre de Puerta

```

//En este recuadro se declara el paquete de la aplicación
package com.example.sensor

//En este recuadro se declara el paquete de la aplicación
import android.Manifest
import android.annotation.SuppressLint
import android.bluetooth.BluetoothAdapter
import android.bluetooth.BluetoothDevice
import android.bluetooth.BluetoothManager
import android.bluetooth.BluetoothSocket
import android.content.Context
import android.content.pm.PackageManager
import android.os.Bundle
import android.widget.ImageButton
import android.widget.Toast
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import java.io.IOException
import java.util.UUID

//En este recuadro se declara la variable bluetoothAdapter y socket
class Abrir_Cerrar : AppCompatActivity() {
    private lateinit var bluetoothAdapter: BluetoothAdapter
    private lateinit var socket: BluetoothSocket

    //Con este código se utilizar para interactuar con el hardware Bluetooth
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_abrir_cerrar)

        val bluetoothManager = getSystemService(Context.BLUETOOTH_SERVICE) as
BluetoothManager
        bluetoothAdapter = bluetoothManager.adapter

//Aquí se indica que el botón Bluetooth es igual a la id Bluetooth si ya tiene
permisos muestra los dispositivos emparejados
        val Bluetooth: ImageButton = findViewById(R.id.Bluetooth)

        //En este recuadro verifica si en el botón Bluetooth se tiene permiso de
bluetooth
        Bluetooth.setOnClickListener {
            if (ContextCompat.checkSelfPermission(this, Manifest.permission.BLUETOOTH) !=
PackageManager.PERMISSION_GRANTED) {
                ActivityCompat.requestPermissions(this,
arrayOf(Manifest.permission.BLUETOOTH), 0)
            } else {
                showPairedDevices()
            }
        }

//Aquí se nombra el botón que se utilizará para abrir la puerta
        val ImageButton4: ImageButton = findViewById(R.id.imageButton4)
        ImageButton4.setOnClickListener {
            sendData("C")
        }

//Aquí se nombra el botón que se utilizara para cerrar la puerta
        val ImageButton: ImageButton = findViewById(R.id.imageButton)
        ImageButton.setOnClickListener {
            sendData("O")
        }
    }
}

```

```

//Este recuadro responde a la solicitud de permiso de bluetooth y muestra los
dispositivos emparejados
override fun onRequestPermissionsResult(requestCode: Int, permissions: Array<out
String>, grantResults: IntArray) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)
    if (requestCode == 0 && grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
        showPairedDevices()
        //Si se deniega el permiso de bluetooth se muestra un mensaje
    } else {
        Toast.makeText(this, "Permiso de Bluetooth denegado",
Toast.LENGTH_SHORT).show()
    }
}
//Este cuadro envia los datos a través del socket y
// si hay un error muestra un mensaje
private fun sendData(data: String) {
    try {
        socket.outputStream.write(data.toByteArray())
    } catch (e: IOException) {
        Toast.makeText(this, "Error al enviar datos", Toast.LENGTH_SHORT).show()
    }
}
//Este cuadro si tiene permisos muestra un cuadro de dialogo con los dispositivos
emparejados
//Cuando se selecciona un dispositivo se conecta a el
@SuppressLint("MissingPermission")
private fun showPairedDevices() {
    if (ContextCompat.checkSelfPermission(this, Manifest.permission.BLUETOOTH) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,
arrayOf(Manifest.permission.BLUETOOTH), 0)
    } else {
        val pairedDevices: Set<BluetoothDevice> = bluetoothAdapter.bondedDevices
        val devicesNames = pairedDevices.map { it.name }.toTypedArray()
        AlertDialog.Builder(this)
            .setTitle("Selecciona un dispositivo")
            .setItems(devicesNames) { _, which ->
                val device = pairedDevices.elementAt(which)
                connectToDevice(device)
            }
            .show()
    }
}
//Si tiene permiso, intenta establecer una conexión con un dispositivo Bluetooth.
// Si hay un error durante este proceso, muestra un mensaje de error.
private fun connectToDevice(device: BluetoothDevice) {
    if (ContextCompat.checkSelfPermission(this, Manifest.permission.BLUETOOTH) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,
arrayOf(Manifest.permission.BLUETOOTH), 0)
    } else {
        try {
            socket =
device.createRfcommSocketToServiceRecord(UUID.fromString("00001101-0000-1000-8000-00805F
9B34FB"))
            socket.connect()
        } catch (e: IOException) {
            Toast.makeText(this, "Error al conectar al dispositivo",
Toast.LENGTH_SHORT).show()
        }
    }
}
}}

```

### 3.4.4 Explicación código de apertura y cierre

- La primera parte del código, es lo mismo que en el código anterior, son librerías que son necesarias para empezar a escribir el código.
- En las siguientes dos líneas de códigos declaramos las variables bluetoothAdapter y socket una vez hecho esto, se ha creado a un código para interactuar con el modem bluetooth.
- Después se ha configurado el botón bluetooth para que de una lista de dispositivos bluetooth vinculados al teléfono móvil solo si tiene permisos.
- Después de haber configurado el bluetooth empezamos a darle funciones a los botones que se encargan de abrir y cerrar la puerta, empezamos nombrando el botón que se encargará de cerrar la puerta después escribiremos una función para que cuando detecte que lo presiones envíe un mensaje que será la letra “C” de Close y se ha hecho lo mismo con el botón de abrir, pero en vez de que envíe el mensaje “C” envíen la letra “O” de Open.
- Empezamos a configurar los permisos, para que puedan responder a la solicitud de permisos para que muestre los dispositivos vinculados de lo contrario si no dan permisos saldrá un mensaje de “Permiso denegado”.
- Después se ha creado un código para enviar los mensajes a través del módem bluetooth y que si hay algún error salga el mensaje de “Error al enviar datos”.
- El penúltimo cuadro de código se encarga de mostrar una tabla con los dispositivos para emparejarse, cuando se selecciona qué dispositivo conectarse, se conecte a él solo si el dispositivo móvil tiene permisos necesarios.
- El último cuadro es el encargado de conectarse al dispositivo bluetooth y si hay algún error al momento de conectarse aparecerá un mensaje “Error al conectar al dispositivo” todo esto lo hace si tiene permiso el dispositivo móvil.

### 3.5 Bot de Telegram

Se ha creado un Bot de Telegram para que avise al propietario cuando se introduzca la contraseña errónea en su domicilio y envíe un mensaje al Telegram del propietario del hogar avisando cada vez que falla un intento.

El código de la imagen es la estructura que extraerá la información **sendTelegramMessage** y al tener esa información la aplicación podrá enviar al bot de Telegram que alguien está intentando entrar a tu casa

```
//En este recuadro se envía un mensaje a un chat de telegram
fun sendTelegramMessage(chatId: String, text: String, botToken: String) {
    CoroutineScope(Dispatchers.IO).launch {
        val urlString =
"https://api.telegram.org/bot$botToken/sendMessage?chat_id=$chatId&text=$text"
        val url = URL(urlString)
        val conn = url.openConnection() as HttpURLConnection
        conn.requestMethod = "GET"
        if (conn.responseCode != 200) {
            println("Failed to send message. Response Code:
${conn.responseCode}")
        }
    }
}
```

En caso de introducir la contraseña de forma errónea el bot de Telegram enviará el mensaje al ID que se te haya asociado para que te enteres de que se ha escrito la contraseña de manera incorrecta.

```
else (sendTelegramMessage("7150768725", "Alguien quiere entrar a tu
casa", "7050245334:AAHY0ws2Dgvgmg7jnFD6-4HwjBHLjLC0IDA"))
```



## 3.6 Página Web

Para la creación de la página web se ha usado Wordpress ya que lo hemos utilizado en la materia de Aplicaciones Web.

Es una página web informativa de nuestra marca y se habla sobre servicios, acerca de nosotros, cómo se puede comprar el servicio, etc.

Para hacer esta página web se ha utilizado:

- Astra es una plantilla de Wordpress que te facilita mucho las cosas a la hora de hacer tu página web, ya que es muy personalizable y tiene infinidad de plantillas para que empieces tu página web con el tema que hayas escogido.
- Elementor es una extensión la cual te permite diseñar tu página web a tu gusto. Es una extensión muy fácil de usar, ya que no se tiene que programar ni nada, ya que todo se hace por bloques y de manera visual.

Para que sean compatibles estas dos herramientas hace falta instalar unas extensiones específicas (como son php-xml, php-curl) para que permita la compatibilidad de la extensión con la plantilla. Para más información sobre cómo se ha configurado se puede acceder al anexo donde se detallan los pasos para seguir adelante

## 4 Estudio económico proyecto

### 4.1 Análisis del Microentorno

#### 4.1.1 Análisis de la competencia

Al hacer un pequeño análisis sobre la competencia, uno de los puntos que nos puede preocupar un poco, son las diversas empresas que hacen puertas correderas, pero como nuestra intención es crear una puerta automatizada dirigida hacia las casas, no nos perjudica tanto, ya que las puertas correderas están dirigidas hacia los establecimientos comerciales o edificios más grandes.

#### 4.1.2 Análisis del sector

El sector de las puertas automatizadas es un sector conocido y usado para tiendas, pero no muy usado para el ámbito de los hogares. Nosotros queremos impulsar este sector dando relevancia a las puertas automatizadas que se abren mediante panel numérico, ya que tienen muchas ventajas respectivamente con las puertas normales que conocemos hoy en día.

#### 4.1.3 Análisis del DAFO

Al ver un poco el sector y la competencia de como está el mercado con las puertas automatizadas hemos realizado un DAFO para explicar las conclusiones que hemos tenido a la hora de hacer la búsqueda de información respecto a otras empresas.

Debilidad	Amenazas
<ul style="list-style-type: none"> <li>● Presupuesto limitado</li> <li>● Somos nuevos en el sector</li> </ul>	<ul style="list-style-type: none"> <li>● Empresas más grandes.</li> <li>● La cantidad de personas que se interesen.</li> </ul>
Fortalezas	Oportunidades
<ul style="list-style-type: none"> <li>● Acuerdo con proveedor para adquirir los productos</li> <li>● Precios competitivos</li> <li>● Tenemos contactos en otros países.</li> </ul>	<ul style="list-style-type: none"> <li>● Comenzar la moda de las puertas automatizadas</li> <li>● Promocionar nuestra marca hasta que se conozca mundialmente</li> </ul>

**Debilidades:** Las debilidades de la empresa es la capital que hay en SensorGate Pro, ya que se hace un gran desembolso económico a la hora de sacar el proyecto hacia delante y como somos nuevos en el sector hay que ir haciendo campañas para llegar a más público que le interese las puertas automatizadas.

**Amenazas:** Las amenazas que se puede tener son las empresas que ya tienen un tiempo con su marca y tiene un prestigio en el mundo de las puertas tecnológicas. Otra amenaza sería que las personas no se interesen de las nuevas tecnologías como puede ser el automatismo de la puerta que nosotros hacemos

**Fortalezas:** Las fortalezas de SensorGate Pro son los contactos que tenemos divididos por todo el mundo, ya que al explicarle la idea a conocidos nuestros mucha gente se ha querido unir al proyecto para crear sedes en diferentes países. Una de las razones por la que conocidos quieren abrir sedes en sus respectivos países es gracias a los precios competitivos que hay en SensorGate Pro y las facilidades que tienen los clientes para consultarnos cualquier duda. Todo esto no se llevaría a cabo sin nuestro proveedor que nos hace descuentos por comprar en grandes cantidades.

**Oportunidades:** Las oportunidades que hay en este sector son grandísimas ya que está en crecimiento exponencialmente la tecnología y si vinculamos la tecnología con las puertas se puede hacer muchísimas cosas que todavía no están desarrolladas.

## 4.2 Idea de Negocio

Nuestra idea de negocio es hacer una puerta automatizada, la cual tiene 2 formas de abrirse y que sean funcionales. La primera forma de abrir la puerta es con un panel numérico, en el que al pulsar un número sale por pantalla. Esta idea con el panel numérico se puede desarrollar más a futuro encriptando la contraseña para que no salga por pantalla la contraseña elegida. La segunda forma es por bluetooth, se abrirá por bluetooth gracias a la aplicación que hemos desarrollado que está vinculada por bluetooth a la puerta automatizada. La aplicación será básica y tendrá un botón de abrir y cerrar la puerta mediante bluetooth. Al hacer cualquiera de las dos opciones habrá un sensor el cual su función será abrir la puerta inmediatamente después de que se haya hecho una de las dos maneras de abrir la puerta. Nuestra misión es darle tranquilidad a los clientes, ya que no tendrán que llevar las llaves y así podrán prevenir disgustos a la hora de perder las llaves, ya que no tendrán.

**Misión:** La misión de SensorGate Pro es dar tranquilidad a nuestros clientes, ya que no tienen que preocuparse nunca más por las llaves. El servicio está enfocado para personas despistadas o con déficit de memoria; que a lo largo del tiempo les viene como anillo al dedo nuestro servicio de puertas automatizadas. También nos gustaría innovar creando diferentes gamas de puertas, priorizando la robustez.

**Visión:** La visión que tenemos en SensorGate Pro es expandir nuestra empresa en todo el mundo, para así ser reconocidos como el mejor servicio de puertas automatizadas. Gracias a nuestros clientes crearemos un antes y un después de las puertas automatizadas.

**Valores:** Los valores que tenemos es que somos un equipo con mucha ambición y ganas de demostrar lo que una empresa pequeña puede hacer y crecer gracias a escuchar y asesorar las críticas de los clientes para mejorar el producto hasta ser uno de los mejores productos del mundo.

## 4.3 Estructura organizativa

### 4.3.1 Planificación RR.HH

En la planificación de recursos humanos de Sensor Gate Pro buscamos jóvenes talentos con ganas de trabajar y sacar la mejor versión de sí mismo.

El plan de Recursos Humanos de SensorGate Pro es el siguiente:

**Búsqueda y Selección de Personal:** Se harán unas preselecciones mediante videos de dos minutos, el cual habrá que explicar la experiencia laboral que tenéis en el ámbito. Cuando ya sean menos los seleccionados se hará presencialmente en nuestro despacho. Al hacer el reclutamiento no se tendrá en cuenta la nacionalidad de los empleados.

**Crecimiento Profesional:** Se establecen unas actividades para desarrollar a los empleados otras virtudes que no conocen como el liderazgo o mejorar algunas que no son su punto fuerte como el trabajo en equipo.

**Valoración y Premios:** Se valorará el trabajo si se acaba antes de tiempo o ayudas a un compañero mediante puntos. 1 punto son 0,10 euros y depende del grado de satisfacción puedes llevarte de 10 puntos hasta 100 puntos dependiendo el desempeño que hayas hecho.

**Estrategias de Talentos:** Se harán unos exámenes de manera seleccionada a los empleados con potencial para mejorar en su ámbito laboral y si sacan buena nota podrán optar ascender de puesto.

**Motivación:** Se harán retos una vez cada 3 meses sobre un temario relacionado con tu cargo y si apruebas el examen se te recompensará con una tarjeta de 100€ en Amazon.

### 4.3.2 Política personal

La Política Personal de SensorGate Pro consiste en:

**Igualdad de derechos:** Todos los empleados tendrán la misma vara de medir para que se les respeten por su trabajo, ya que cada uno se especializa en un sector

**Clima laboral:** Se han de respetar las normas de la empresa para que haya una comunicación limpia y cada trabajador tenga su espacio para trabajar tranquilamente.

**Protocolo de Resolución de Problemas:** Para los problemas tenemos un protocolo que es mediante un intermediario que lo que hace es ayudar

**Estructura y organización:** La estructura de la empresa se ha de respetar los roles que tiene cada una de las personas para prevenir malos entendidos.

**Salud:** En SensorGate Pro nos preocupamos por la salud mental de los empleados y lo que hacemos son ejercicios para trabajar la salud mental de los trabajadores.

**Desconexión:** En cada hora de trabajo tienen 10 minutos para salir a tomar el sol y a estirar las piernas para que desconectes esos minutos del trabajo.

## 4.4 Estrategia de Marketing

Uno de los puntos más importantes son los objetivos como empresa ya que para crecer y seguir adelante necesitamos encontrar objetivos a corto y largo plazo.

Nuestros objetivos a corto plazo es expandir la marca y que se hable más para que más gente la conozca.

Nuestra estrategia de Marketing consiste en hacer anuncios y contratar a instagramers y Tiktokers de todo Europa que tendrán un código del 10% de descuento y lo subirán a sus respectivas redes sociales. Si los clientes compran puertas automatizadas con el código del influencer les saldrá más barata la puerta y nos ayudará a expandirnos en el sector, comenzaremos en el décimo mes haciendo las campañas ya que meses anteriores estaremos contactando y planificando los siguientes meses, cada mes colaboraremos con personas diferentes iremos a su casa y grabaremos contenido para nuestras redes sociales explicando las ventajas que tiene nuestra puerta automatizada respecto a las puertas normales en tema de seguridad y privacidad.

## 4.5 Plan jurídico fiscal

Al hacer una investigación de cada forma judicial y lo que ofrecen nos hemos decantado por tener una fórmula jurídica de sociedad limitada para SensorGate Pro por lo siguientes puntos.

Uno de los puntos que nos han hecho decantarme ha sido el tema de los socios, ya que cada socio tiene una importancia dependiendo del dinero aportado a la empresa y se puede tener varios socios. También otra cosa que nos ha hecho decantarnos ha sido que se necesita menos dinero para hacer una Sociedad limitada que una sociedad anónima y es más flexible.

Si todo empieza ir bien y abrimos sedes en diferentes países nuestra intención es cambiar a sociedad anónima si la ley lo permite y acepta nuestra documentación.

## 4.6 Plan económico financiero

### 4.6.1 Plan de inversiones iniciales

Los fondos de nuestro plan de inversiones iniciales, con 3 socios, son los siguientes:

- **Inversión inicial:** Cada socio hace una inversión de 2000€ y se quita el 50% de su sueldo para sacar la empresa adelante. Somos un total de 3 socios así que serán 600€.
- **Materiales:** Se destinan 300€ para comprar los materiales que se necesitan para hacer la puerta automatizada, cada 300€ es una puerta automatizada hecha.
- **Alquiler del almacén:** El local que hemos encontrado es de 300m<sup>2</sup> y ha costado 300€ al mes.
- **Fianza del almacén:** La fianza del almacén es de 600€ que son dos meses de alquiler del local.
- **Marketing:** Se depositará 600€ al mes a partir del segundo mes para hacer publicidad con instagramers y tikkers. El primer mes es para contactar con los influencers y llegar a un acuerdo.
- **Furgoneta:** La furgoneta nos ha costado alquilarla 550€ por mes y la alquilamos a partir del segundo mes, al alquilarla ya viene todo el seguro incluido.
- **Autónomos:** Se destinan unos 1320€ al año para pagar lo que es ser autónomo los primeros años, cada año van subiendo el precio.

#### 4.6.2 ROI (retorno de la inversión)

Para calcular el ROI hay que hacer la siguiente operación:

$$\text{ROI} = ((\text{Ingresos} - \text{Inversión}) / \text{Inversión}) \times 100$$

$$\text{ROI(2024)} = (69900 - 21000) / 21000 \times 100 = 232,86\%$$

$$\text{ROI(2025)} = (108000 - 25800) / 25800 \times 100 = 318,60\%$$

#### 4.6.3 Previsión de ingresos y gastos

PLAN DE INGRESOS Y GASTOS	Mes 1	Mes 2	Mes 3	2024	2025
Ingresos	0 €	1.260 €	2.520 €	69.300 €	108.000 €
Gastos directos	0 €	600 €	1.200 €	33.000 €	63.000 €
Margen de Contribución	0 €	660 €	1.320 €	36.300 €	45.000 €
Gastos Indirectos	1.011 €	2.161 €	6.661 €	54.782 €	63.782 €
Sueldos	0 €	0 €	3.000 €	30.000 €	39.600 €
Reinversión del 50% del Sueldo	0 €	0 €	1.500 €	15.000 €	19.800 €
Autónomos	110 €	110 €	110 €	1.320 €	1.320 €
Furgoneta	0 €	550 €	550 €	6.050 €	6.050 €
Fianza Almacén	600 €	0 €	0 €	600 €	0 €
Almacén	301 €	301 €	301 €	3.612 €	3.612 €
Combustible	0 €	600 €	600 €	6.600 €	6.600 €
Marketing	0 €	600 €	600 €	6.600 €	6.600 €
Recuento total	-1.011 €	-1.501 €	-3.841 €	-3.482 €	1.018 €

## 5 Problemas a la hora de hacer el proyecto

En este proyecto han surgido varios problemas durante la instalación y montaje. A continuación, os voy a contar qué impedimentos nos han surgido y cómo los hemos solucionado, o cómo hemos cambiado el rumbo cuando no hemos encontrado solución.

- El primer problema que nos ha surgido ha sido con la página web que creamos con Wordpress. Estuvimos semanas buscando porque nos salía el error de JSON si habíamos instalado las dos extensiones que eran el php-xml y el php-curl y también ejecutamos el comando **sudo a2enmod rewrite** para que Wordpress pueda leer archivos .htaccess. Después de estar semanas buscando información relacionada con el tema encontramos un foro el cual nos decía la solución, que era entrar en el archivo de configuración de **/etc/apache2/apache2.conf** y modificar el directorio **/var/www/** de la siguiente forma:

```
<Directory /var/www/>
Options Indexes FollowSymLinks
AllowOverride all # —> (Por defecto viene en None)
Require all granted
</Directory>
```

Ha fallado la actualización. Las respuesta no es una respuesta JSON válida.

- Otro de los problemas que nos han surgido en este proyecto ha sido a la hora de hacer la aplicación con Android Studio ya que muchos de los plugins que hemos instalado no admitía la versión más reciente, después de estar informándonos del error encontramos la página de sonatype que es un sistema de gestión de repositorios en el cual se almacenan dependencias y encontramos la solución. Teníamos que poner la versión 1.6.2 para que se desbloquearan paquetes nuevos para la comunicación a internet.
- A la hora de hacer la aplicación nos han surgido muchos problemas, ya que no había mucha información para buscar. El problema que hemos tenido ha sido a la hora de enviar un SMS o correo electrónico en caso de escribir mal la contraseña, ya que habían muchos videos distintos pero ninguno funcionaba correctamente. Con este error estuvimos la mayoría de los meses ya que pensábamos que lo íbamos a solucionar y al estar a 1 mes de la entrega decidimos cambiar en vez de que envíe un correo o SMS, lo que haremos es que nos avise mediante Telegram gracias al bot de seguridad.



## 6 Conclusiones

### 6.1 Conclusiones generales del proyecto

Hacer el proyecto nos ha ayudado a entendernos mejor a la hora de trabajar en grupo y a abrir nuestras mentes y conocer conocimientos que no habíamos tocado como la programación con los diferentes lenguajes que hemos tocado.

Se ha hecho un buen trabajo ya que se han cumplido nuestros objetivos del proyecto aunque hemos pasado épocas en las que nos hemos visto un poco atascado por falta de información que había sobre nuestro proyecto

La conclusión que se ha hecho es que se ha aprendido a crear una puerta automatizada que se pueda abrir mediante panel numérico o app y el proyecto tiene muchas formas de profesionalizarse y ponerlo en marcha en la vida real

### 6.2 Consecución de los objetivos

Los objetivos que nos habíamos puesto al principio de la memoria los hemos cumplido en creces gracias al esfuerzo que hemos dado al proyecto.

Se ha logrado desarrollar una aplicación móvil programando que era nuestra primera vez y la hemos hecho compatible para que se pueda vincular mediante bluetooth y en caso de escribir el pin de forma incorrecta llegará un aviso al telegram. Todo esto parece fácil pero es un camino que nos ha costado y ha durado mucho tiempo para encontrar la configuración adecuada. También se ha aprendido a programar en Arduino que no lo habíamos tocado mucho y se ha implementado todos los componentes para que vayan a la vez y estén conectados entre sí.

### 6.3 Valoración de la metodología y planificación

La planificación ha sido clara y se ha intentado seguir lo máximo posible ya que si estábamos en lo previsto según el diagrama de gantt eso significaba que íbamos bien de tiempo. La metodología de trabajo se ha seguido bastante bien ya que habíamos elegido el método de cascada que es seguir un objetivo de la tarea hasta cumplirlo sin pasar al siguiente y a la vez nos ha hecho estar mucho tiempo en el código de la aplicación de Android Studio y en algún momento hemos tenido que empezar otras cosas y dejarlo de lado porque estábamos atascados y no se nos ocurrían soluciones. La clave del proyecto ha sido desconectar un poco del desarrollo de la aplicación móvil y hacer otros objetivos que se les estaba dando menos prioridad.

## 6.4 Visión de futuro

En un futuro nuestra misión es mejorar la seguridad de la puerta automatizada para que sea robusta e impenetrable. También nos gustaría mejorar la parte visual de la aplicación móvil implementando temas para que puedas elegir qué fondo quieres en la página del pin. Otra idea de futuro sería de hacer la página web que sea funcional en el proyecto para que la gente se pueda registrar y si has comprado nuestro servicio te salga la aplicación para descargarla y te pida que pin quieres que se te asigne como contraseña.

## 8 Webgrafía

CC BY-NC-ND 3.0 ES Deed | Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya | Creative Commons. (s. f.).

<https://creativecommons.org/licenses/by-nc-nd/3.0/es/deed.ca>

Arduino Bot. (s.f.). Manual de Programación Arduino. Recuperado de <https://arduinoobot.pbworks.com/f/Manual+Programacion+Arduino.pdf>

Bluetooth Service for Android with Kotlin. (s. f.). Gist.

<https://gist.github.com/dcalsky/f151cebdfe66e4b4e1d774e497f83953>

Maven Central: org.lucee:javamail. (s. f.). Maven Central.

<https://central.sonatype.com/artifact/org.lucee/javamail>

Garaje de ideas, Tech. (2022, 10 agosto). Botones paso a paso en Android Studio | Kotlin [Video]. YouTube. <https://www.youtube.com/watch?v=AnFQToGSaPY>

CodeLab. (2022, 13 febrero). Varias pantallas en Android Studio utilizando KOTLIN | Crear Botón para acceder a otra Pantalla [Video]. YouTube.

<https://www.youtube.com/watch?v=NSqORENpaTM>

Programación Android by AristiDevs. (2020, 7 octubre). TUTORIAL Permisos en ANDROID con KOTLIN [Video]. YouTube. <https://www.youtube.com/watch?v=rdfjT0bQBgs>

Deccon Tech. (2023, 19 marzo). Bluetooth Permission and enable Bluetooth | Android Studio KOTLIN [Video]. YouTube. [https://www.youtube.com/watch?v=\\_bMK5lwx-as](https://www.youtube.com/watch?v=_bMK5lwx-as)

YouTube. (2023, May 15). AI and the Future of Work [Video]. YouTube.

<https://www.youtube.com/shorts/4rG6JRCZljw>

Llamas, L. (2016, 2 octubre). Usar un teclado matricial con Arduino. Luis Llamas.

<https://www.luisllamas.es/arduino-teclado-matricial/>

Tutorial LCD con I2C, controla un LCD con solo dos pines. (s. f.). Naylamp Mechatronics - Perú.

[https://naylampmechatronics.com/blog/35\\_tutorial-lcd-con-i2c-controla-un-lcd-con-solo-dos-pines.html](https://naylampmechatronics.com/blog/35_tutorial-lcd-con-i2c-controla-un-lcd-con-solo-dos-pines.html)

Tutorial de Arduino y sensor ultrasónico HC-SR04. (s. f.). Naylamp Mechatronics - Perú.

[https://naylampmechatronics.com/blog/10\\_tutorial-de-arduino-y-sensor-ultrasonico-hc-sr04.html](https://naylampmechatronics.com/blog/10_tutorial-de-arduino-y-sensor-ultrasonico-hc-sr04.html)

Cómo crear y conectar un chatbot de Telegram | SendPulse. (s. f.). SendPulse.

<https://sendpulse.com/latam/knowledge-base/chatbot/telegram/create-telegram-chatbot>

Copyright © 2024 SensorGate\_Pro.

Este documento está licenciado bajo la licencia Creative Commons Reconocimiento-NoComercial-SinObraDerivada 3.0 España (CC BY-NC-ND 3.0 ES).

Puede copiarlo y compartirlo con atribución, pero no puede usarlo comercialmente ni crear obras derivadas.

## 9 Anexos

Link de Android Studio: <https://developer.android.com/studio>

Link de WordPress: <https://wordpress.com/es/>

Link de Arduino: <https://www.arduino.cc/en/software>

Link Wordpress problema con el Json:

<https://es.wordpress.org/support/topic/mensaje-de-error-las-respuesta-no-es-una-respuesta-json-valida/page/7/>