



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE RECONOCIMIENTO
FACIAL MEDIANTE P5.JS Y TEACHABLE MACHINE PARA LA APERTURA DE UNA
PUERTA**

Trabajo de titulación previo a la obtención del
Título de Ingeniero/a en Electrónica y Automatización

AUTORES: BRYAN ANTHONY JUANAZO ARRIAGA
MICHELLE XIOMARA LAMBOGGLIA YAR
TUTOR: ING. RAFAEL FRANCO REINA, MSC.

Guayaquil – Ecuador
2023 -2024

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros Lamboggia Yar Michelle Xiomara con documento de identificación N° 0953530474, y Juanazo Arriaga Bryan Anthony con documento de identificación N° 0941498727; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación

Guayaquil, 29 de enero del año 2024.

Atentamente,



Bryan Anthony Juanazo Arriaga
CI: 0941498727



Michelle Xiomara Lamboggia Yar
CI: 0953530474

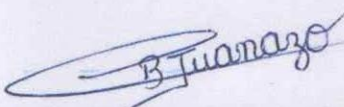
**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Lamboggia Yar Michelle Xiomara con documento de identificación N° 0953530474 y Juanazo Arriaga Bryan Anthony con documento de identificación N° 0941498727, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: “Diseño e Implementación de un Sistema de Reconocimiento Facial mediante P5.JS y Teachable Machine para la apertura de una puerta”, el cual ha sido desarrollado para optar por el título de: Ingeniero/a en Electrónica, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 29 de enero del año 2024.

Atentamente,


Bryan Anthony Juanazo Arriaga
CI: 0941498727


Michelle Xiomara Lamboggia Yar
CI: 0953530474

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Rafael Christian Franco Reina con documento de identificación N° 0923328629, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE RECONOCIMIENTO FACIAL MEDIANTE P5.JS Y TEACHABLE MACHINE PARA LA APERTURA DE UNA PUERTA, realizado por Juanazo Arriaga Bryan Anthony con documento de identificación N° 0941498727 y Lamboggia Yar Michelle Xiomara con documento de identificación N° 0953530474, obteniendo como resultado final el trabajo de titulación bajo la opción de Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 29 de enero del año 2024.

Atentamente,



Ing. Rafael Christian Franco Reina, MSc.

C.I.: 0923328629

DEDICATORIA

En primer lugar, este trabajo es para toda mi familia, a mis padres Julissa y Augusto quienes siempre estuvieron apoyándome en mi camino y crecimiento a lo largo de esta carrera universitaria la cual muchas veces me sentí que ya no podía, pero por sus palabras de aliento nunca dejaron que me rinda, además por siempre enseñarme valores los cuales me hacen una persona de bien.

A mi hermana July Noemi, la cual es mi inspiración y por la cual lucho día a día para ser un ejemplo para ella cuando le toque cursar sus estudios superiores

Bryan Anthony Juanazo Arriaga

A Dios dedico este trabajo, por brindarme buena salud para culminar con éxito mi trayecto en la universidad, a mis padres que estuvieron conmigo dándome su ayuda y su reflexión para hacer de mí una persona de bien y llegar a esta estancia de mis estudios.

A mis tíos y primos, que me aconsejaron y ayudaron durante mi carrera, que me dieron fuerzas para no rendirme y llegar hasta el final.

Michelle Xiomara Lamboggia Yar

AGRADECIMIENTO

Doy gracias a Dios por ayudarme a cumplir cada uno de los objetivos planteados en mi formación académica y profesional.

A mis padres por siempre darme su apoyo cuando más lo necesite y por siempre estar ahí para escucharme y aconsejarme.

A los docentes quienes nos brindaron sus conocimientos y experiencias en cada clase impartida para así formarnos y prepararnos como buenos profesionales.

Bryan Anthony Juanazo Arriaga

Agradezco a Dios por haberme brindado vida y fortaleza para superar las dificultades durante mi enseñanza académica, además de concederme una familia admirable.

A mis padres por creer en mí, enseñándome a valorar todo lo que tengo, a alcanzar con éxito mis metas, por darme su amor y educación para ser la persona que soy ahora.

A mi familia, que de una u otra forma han estado en mi vida, en momentos de felicidad, tristeza y solidaridad.

A mis compañeros y amigos que estuvieron ahí siendo un apoyo fundamental, dándome su amistad incondicional y compañerismo.

Michelle Xiomara Lamboggia Yar

Resumen

En este proyecto se implementó un sistema de reconocimiento facial que utiliza por base de datos un programa de Google con nombre Teachable Machine; programación en el entorno P5.JS y la librería de JavaScript; SerialPort como aplicación para la conexión en el programa P5.JS y el Arduino; y programación de salida serial en el Arduino UNO para la cerradura.

Este proyecto se desarrolla estableciendo la base de datos a utilizar, en la que se crea un modelo de clasificación de imágenes con distintas clases, con el cual se pueda realizar el reconocimiento facial; para esto se prepara una codificación en el entorno p5.js, por medio del cual se exporta el link de la base de datos y se especifica el puerto del Arduino a usar, en la aplicación del microcontrolador, se establece a que usuario se le permite el acceso a la habitación y a que usuario se le restringe el paso.

Los resultados de esta investigación muestran la versatilidad del programa Teachable Machine al tener múltiples funciones, tales como: modelos de clasificación de imágenes, sonidos y posturas.

En el desenlace de este prototipo, se muestra como con conocimientos básicos en Teachable Machine y P5.JS, se puede diseñar un sistema de reconocimiento facial para el control de acceso a una habitación.

Palabras Clave: Reconocimiento Facial, Teachable Machine, P5.JS, Inteligencia Artificial, Arduino, SerialPort, JavaScript.

Abstract

In this project, a facial recognition system was implemented that uses a Google program called Teachable Machine as a database; programming in the P5.JS environment and the JavaScript library; SerialPort as an application for connecting the P5.JS program and the Arduino; and serial output programming on the Arduino UNO for the lock.

This project is developed by establishing the database to be used, in which an image classification model with different classes is created, with which facial recognition can be performed; For this, coding is prepared in the p5.js environment, through which the database link is exported and the Arduino port to be used is specified. In the microcontroller application, it is established which user is allowed. access to the room and which user is restricted.

The results of this research showed the versatility of the Teachable Machine program by having multiple functions, such as: image, sound and posture classification models.

In the outcome of this prototype, it is shown how with basic knowledge in Teachable Machine and P5.JS, a facial recognition system can be designed to control access to a room.

Keywords: Facial Recognition, Teachable Machine, P5.JS, Artificial Intelligence, Arduino, SerialPort, JavaScript.

ÍNDICE DE CONTENIDO

I	INTRODUCCIÓN.....	1
II	PROBLEMA.....	2
III	OBJETIVOS	3
3.1	Objetivo general	3
3.2	Objetivos específicos.....	3
IV	FUNDAMENTO TEÓRICO	4
4.1	Teachable Machine	4
4.2	TensorFlow.....	5
4.3	P5.JS.....	5
4.4	Biblioteca P5.js y aplicación P5.serialcontrol.....	6
4.5	Arduino.....	7
4.6	Reconocimiento Facial.....	8
4.7	Inteligencia Artificial	9
V	MARCO METODOLÓGICO.....	11
5.1	Diseño esquemático del proyecto.....	12
5.2	Diseño del modelo de clasificación de imágenes en Teachable Machine.....	13
5.2.1	Selección del proyecto deseado en Teachable Machine.	13
5.2.2	Diseño del algoritmo en el entorno P5.js.	16

VI	RESULTADOS.....	24
6.1	Modelo de imagen como base de datos en Teachable Machine	24
6.2	Algoritmo en P5.JS y Arduino.....	26
6.3	Prueba y Error en la etapa de evaluación del sistema	27
6.4	Prueba final del sistema.....	28
6.5	Análisis de datos según la matriz de confusiones de Teachable Machine	31
VII	CRONOGRAMA.....	32
VIII	PRESUPUESTO	34
IX	CONCLUSIONES	35
X	RECOMENDACIONES.....	36
XI	REFERENCIAS.....	37
XII	ANEXO.....	39

Índice de Figuras

<i>Figura 1. Teachable Machine - clasificación de imagen.</i>	4
<i>Figura 2. Esquema de TensoFlow con Teachable Machine</i>	5
<i>Figura 3. Edición de elipse mediante el mouse</i>	6
<i>Figura 4. Diagrama de la conexión desde el puerto serie a p5.js a través de p5.serialcontrol</i>	7
<i>Figura 5. Arduino UNO</i>	8
<i>Figura 6. Reconocimiento por rostro</i>	9
<i>Figura 7. Inteligencia Artificial</i>	10
<i>Figura 8. Esquema general del proceso de implementación del sistema.</i>	11
<i>Figura 9. Diseño esquemático.</i>	12
<i>Figura 10. Selección de un nuevo proyecto Teachable Machine</i>	13
<i>Figura 11. Selección del modelo a escoger para la clasificación de imagen</i>	14
<i>Figura 12. Creación de las clases a usar.</i>	14
<i>Figura 13. Preparación del modelo.</i>	15
<i>Figura 14. Sección para escoger la forma de exportar el modelo.</i>	15
<i>Figura 15. Variables a usar y función de precarga para el video de la cámara.</i>	16
<i>Figura 16. Función setup de P5.JS.</i>	17
<i>Figura 17. Función creada para la obtención de resultados.</i>	17
<i>Figura 18. Nombres de usuarios.</i>	18
<i>Figura 19. Visualización de usuarios y datos en la consola.</i>	18
<i>Figura 20. Sección de librerías y descripción de funcionamiento.</i>	19
<i>Figura 21. Aplicación p5.serialcontrol.</i>	20
<i>Figura 22. Variables a usar y función setup.</i>	21
<i>Figura 23. Void loop, asignación de pines, etapa 1.</i>	21

<i>Figura 24. Void loop, asignación de pines, etapa 2.</i>	22
<i>Figura 25. Void loop, asignación de pines, etapa 3.</i>	22
<i>Figura 26. Void loop, asignación de pines, etapa 4.</i>	23
<i>Figura 27. Comprobación de datos en Teachable Machine.</i>	24
<i>Figura 28. Registro de datos.</i>	25
<i>Figura 29. Preparación del modelo.</i>	25
<i>Figura 30. Prueba de algoritmo en P5.JS.</i>	26
<i>Figura 31. Conexión en protoboard.</i>	26
<i>Figura 32. Prueba y Error.</i>	27
<i>Figura 33. Pruebas en la maqueta.</i>	28
<i>Figura 34. Pruebas para apertura de un usuario.</i>	29
<i>Figura 35. Visualización de la maqueta final.</i>	29
<i>Figura 36. Parte interna de la maqueta.</i>	30
<i>Figura 37. Visualización de maqueta completa.</i>	30
<i>Figura 38. Matriz de confusión generada</i>	31
<i>Figura 39. Diagrama de Gantt del cronograma del proyecto.</i>	32

Índice de Tablas

<i>Tabla 1.</i>	33
<i>Tabla 2.</i>	34

I INTRODUCCIÓN

Debido al gran avance en la ciencia tecnológica, el uso de dispositivos electrónicos forma parte del día a día, en distintos ámbitos, ya sea, personal, recreativo, domestico, etc. En la época actual, la tecnología abre escenarios que hace años atrás se podían describir como ciencia ficción, uno de estos ejemplos es el reconocimiento facial el cual se ha transformado en un instrumento primordial en sectores donde se requiera seguridad (Fernández, 2019).

El presente trabajo de Titulación se sumerge en el fascinante mundo del sistema de reconocimiento facial, en la cual, hará uso de la misma para crear un sistema embebido, para ser utilizado en el hogar. En este sistema se crea una base de datos en Teachable Machine usando clasificación de imágenes, por lo que es necesario, establecer un numero de clases a usar.

El desarrollo de este trabajo es realizado mayormente en el entorno de programación libre P5.JS, que hace uso de librerías de JavaScript; A su vez, se utiliza la librería p5.Serialport y la aplicación p5.serialcontrol, para la conexión del P5.JS al Arduino, en el que se establece los pines a usar para el encendido de los diodos led, al igual que la apertura de la cerradura.

Este proyecto muestra como un sistema de reconocimiento facial, aporta positivamente a la seguridad en el hogar, se enfoca principalmente a la privacidad del cuarto de cualquier miembro de la familia, así el miembro que haga uso de este sistema de reconocimiento facial puede especificar que otros miembros tienen acceso a su habitación y quienes no.

II PROBLEMA

La privacidad es imprescindible y una razón para proteger la vida privada de los individuos, debido a que cada uno gestiona su intimidad a su gusto y siempre hay cosas de las cuales no se quiere mostrar a nadie, y es preciso mantenerlo en un lugar donde solo esa persona tenga acceso, en este caso su propia habitación (Aldundia, 2021).

Se sabe que en un hogar hay muy poca privacidad entre los miembros, ya que como máximo el tocar la puerta antes de entrar sería una de las normas de respeto a la privacidad en el hogar, por eso muchos miembros optan por ponerle seguro a las puertas de sus cuartos, lo que con lleva en una llamada de atención por parte de la madre o el padre.

Por esta razón es necesario diferenciar los distintos tipos de intimidad que existen; intimidad informativa la cual se refiere a la confidencialidad en relación con el derecho a la custodia de las informaciones personales, intimidad espacial la cual se refiere al espacio que forma el ámbito privado, es de vital importancia garantizar el cuidado de las relaciones sociales y el progreso de las acciones privadas y por último la intimidad referida a la personalidad que hacen mención al acceso limitado y el trato respetuoso de los diferentes aspectos que forman la personalidad del individuo (Maite, 2021).

III OBJETIVOS

3.1 Objetivo general

Diseñar e implementar un sistema de reconocimiento facial mediante P5.js y Teachable Machine para la apertura de una puerta.

3.2 Objetivos específicos

- Realizar un modelo de clasificación de imágenes utilizando Teachable Machine que permita el reconocimiento facial.
- Programar en el entorno p5.js un algoritmo que permita conectarse con Arduino para controlar el proceso de bloquear y desbloquear una puerta.
- Evaluar el sistema de reconocimiento facial en tiempo real mediante el análisis de datos previamente ingresados para medir la fiabilidad y el tiempo de respuesta del sistema.

IV FUNDAMENTO TEÓRICO

4.1 Teachable Machine

Teachable machine es un instrumento fundamentado en la web, en el cual es posible establecer modelos de aprendizaje automático de forma ágil, fácil y factible (Google, Conceptos Básicos, 2019).

El proyecto de Google Teachable Machine acerca la IA (Inteligencia Artificial) a las personas que desean experimentar con la misma, no se ocupa conocer de programación. Esta aplicación permite reconocer sonidos, posturas e imágenes, las cuales, mediante clasificación de modelos son cargadas por el usuario (UTEC, 2020).

En la figura 1, se realiza la clasificación de imágenes ya sea con una foto o video en vivo mediante la cámara web.

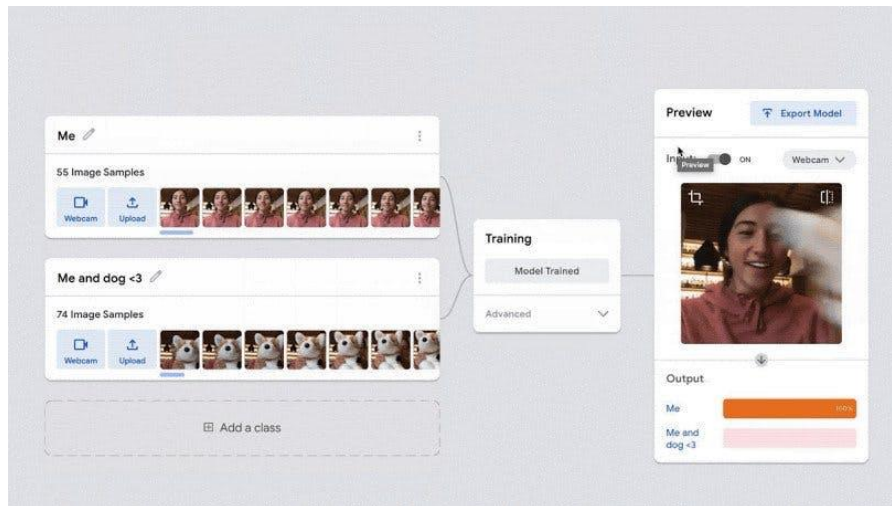


Figura 1. Teachable Machine - clasificación de imagen (Google, Teachable Machine, 2017).

Teachable Machine usa Tensorflow.js, un librero de aprendizaje automático en JavaScript. Permite preparar y efectuar modelos que se crean en un navegador web.

4.2 TensorFlow

Es un librero de estudio de código abierto que puede efectuarse en cualquier lugar donde JavaScript pueda. Se basa en el librero original de TensorFlow escrita en Python y tiene como función recrear experiencia de desarrollador y el conjunto de API para el ecosistema de JavaScript (Google, 2022)

Dado la portabilidad de JavaScript, se puede utilizar en los siguientes entornos:

- Perspectiva del cliente en la web usando JavaScript básico.
- Perspectiva del servidor y dispositivos IoT como Raspberry Pi usando Node.js.
- Aplicaciones de interfaz usando Electron.
- Aplicaciones móviles originarias que usan React Native.

En la figura 2, se observa la interfaz de TensorFlow con Teachable Machine.

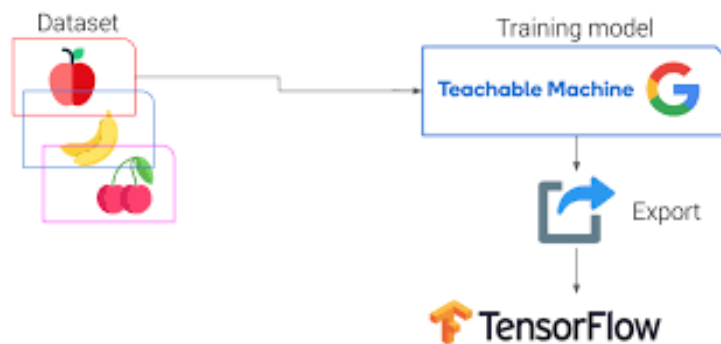


Figura 2. Esquema de TensorFlow con Teachable Machine (Techzizou, 2022).

4.3 P5.JS

Es un librero de JavaScript para todo tipo de programación de cualquier área. P5.js es muy eficiente ya que es una multiplataforma y es compatible con varios hardware y navegadores web (Draz, 2022).

La ventaja de usar el lenguaje JavaScript es su amplia disponibilidad y soporte, además, el procesamiento es ese lenguaje que enfatiza la viabilidad de que programadores creen prototipos de software rápidamente (Sarthak, 2021).

Se puede observar en la figura 3, un ejemplo de P5.js, sobre la edición de una elipse.

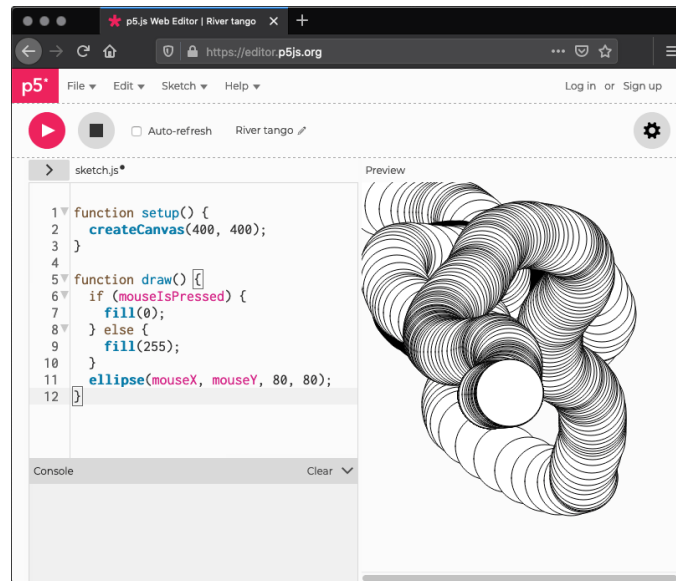


Figura 3. Edición de elipse mediante el mouse (Sarthak, 2021).

Diferencia entre P5.JS y JavaScript

JavaScript es un lenguaje básico que proporciona todas las características para incorporar cualquier funcionalidad en los navegadores. Puede usar bucles, condicional, etc.

P5.JS es una biblioteca de JavaScript. P5.JS se ejecuta en JavaScript puro y proporciona algunas funciones que facilitan al usuario en el navegador (Sarthak, 2021).

4.4 Biblioteca P5.js y aplicación P5.serialcontrol

La biblioteca P5.js serialport y la aplicación p5.serialcontrol, se usa para la comunicación con el microcontrolador en serie. La biblioteca P5.js serialport no puede acceder directamente a los puertos serie cuando se ejecuta un boceto en un navegador porque el navegador no tiene acceso

directo al puerto serie. Pero puede comunicarse con otro programa en su computadora que puede intercambiar datos con el puerto de serie. p5.serialcontrol es la aplicación que conecta el boceto, que se ejecuta en un navegador, con los puertos serie del equipo, Figura 4 (NYU, 2020).

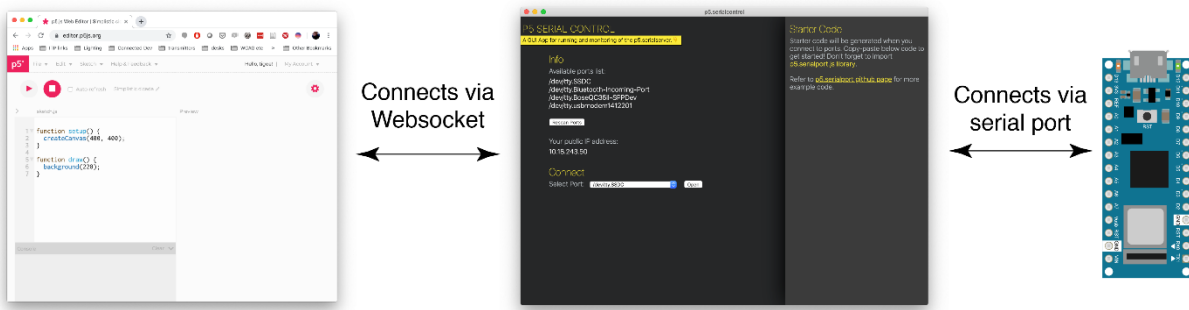


Figura 4. Diagrama de la conexión desde el puerto serie a p5.js a través de p5.serialcontrol (NYU, 2020).

4.5 Arduino

Arduino se utiliza para el desarrollo de elementos autónomos, o también, se conecta a otros dispositivos, se comunica con otros programas, para la interacción del hardware y el software.

Un ejemplo claro puede ser el control de un elemento, como un motor que suba o baje una persiana acorde a la luz detectada gracias a un sensor conectado al Arduino, o también, para convertir la información de una fuente, como un teclado, y transformar la información a algo que comprenda, en concreto, un ordenador (Zambetti, 2020).

El uso de Arduino no se puede catalogar en dos grupos actualmente:

- Arduino se usa como un microcontrolador, cuando se descarga un programa desde el ordenador y se ejecuta de manera independiente de éste, y dirige y sustenta dispositivos específicos y basa sus decisiones conforme al programa descargado y se relaciona con el mundo físico por los sensores y actuadores.
- Arduino hace de interconexión entre un ordenador y distintos dispositivos, que realiza una tarea específica, para interpretar dicha tarea en el mundo físico a una

acción, o inversamente, debido a sensores que se encuentran conectados a la placa Arduino se puede lograr que el ordenador realice una acción específica.

En la figura 5, se muestra el Arduino Uno, el cual es una buena opción para iniciar en la programación.

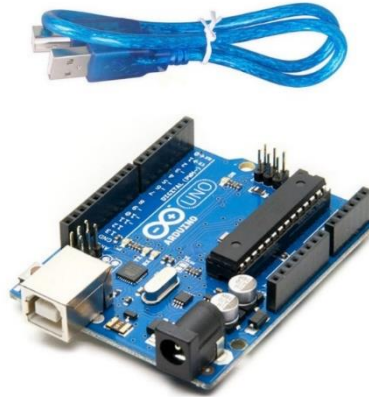


Figura 5. Arduino UNO (Mecatronica, 2024).

4.6 Reconocimiento Facial

Es una ciencia tecnológica, apto para equiparar o comprobar a un individuo por medio de una imagen, vídeo de su rostro. Usualmente, esta confirmación es usada para conceder el acceso a una aplicación, sistema o servicio (ID, 2022).

El detector facial reconoce la ubicación de cada imagen y luego el clasificador detecta la cara. El clasificador realiza el aprendizaje automático al entrenar la imagen varias veces, para detectar la cara en la base de datos (Ashwini, Balaji, Srinivasan, & Monisha, 2018).

Es una manera de asociación biométrica que se basa de medidas corporales, en esta ocasión la cara y cabeza, para corroborar la identidad de un individuo por medio de su patrón biométrico y datos adicionales, Figura 6.

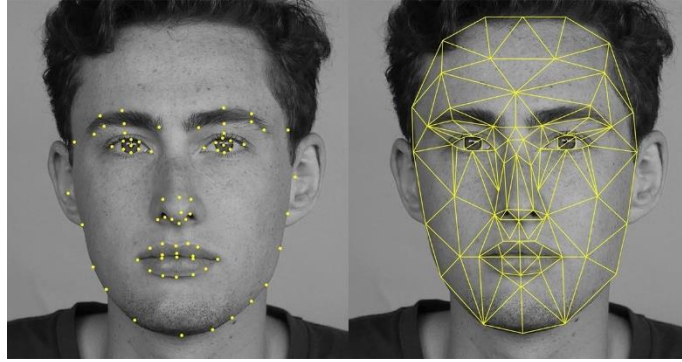


Figura 6. Reconocimiento por rostro (Lacort, 2017).

4.7 Inteligencia Artificial

Citando a la comisión europea la inteligencia artificial se describe como método de software (y probablemente de hardware también) bosquejado por seres humanos que, frente a un objetivo complejo, se muestra en la dimensión física o digital:

- Captando su alrededor, por medio de la obtención y deducción de datos estructurados o no estructurados.
- Argumentando sobre el conocimiento, elaborando la información proveniente de estos datos y eligiendo las mejores operaciones para conseguir el objetivo dado.

Las inteligencias artificiales se caracterizan por emplear algoritmos y modelos matemáticos para elaborar una extensa cantidad de datos y apropiarse de decisiones fundamentadas en patrones y pautas establecidas por medio del aprendizaje automático, lo cual se establece en la capacidad de una máquina para instruirse de forma automática por medio de datos sin ser programada particularmente para hacerlo. De esta forma la inteligencia artificial mejora su exactitud y eficacia con el tiempo, Figura 7 (España, 2023).

Existen dos tipos de inteligencia artificial:

- **Software:** ayudantes virtuales, estudio de imágenes, motores de indagación o métodos de reconocimiento de voz y rostro.
- **Inteligencia Artificial Integrada:** autómatas, drones, vehículos independientes o el internet de las cosas.

La inteligencia artificial en el área de seguridad representa la próxima etapa en la defensa cibernética. El aumento en la programación malintencionada y otras amenazas produce que la ciberseguridad tradicional sea posiblemente más compleja. La IA cubre la carencia de talentos que forma vacantes de trabajo en ciberseguridad. También, se plantea la inteligencia artificial como una solución escalable para los instrumentos de seguridad, a medida que crece el flujo de trabajo. (servicenow, 2024).



Figura 7. Inteligencia Artificial (España, 2023).

V MARCO METODOLÓGICO

Este proyecto se realizó mediante una metodología de carácter exploratorio debido a que se obtuvo información de los materiales y datos para el análisis del reconocimiento facial propuesto. Además del levantamiento de información de software, aplicaciones, librerías, en este caso, P5.JS, Teachable Machine y Arduino, y el correcto uso de cada una, esto incluye una metodología cuantitativa para la comparación de software CAD y herramientas tecnológicas en el proceso de investigación.

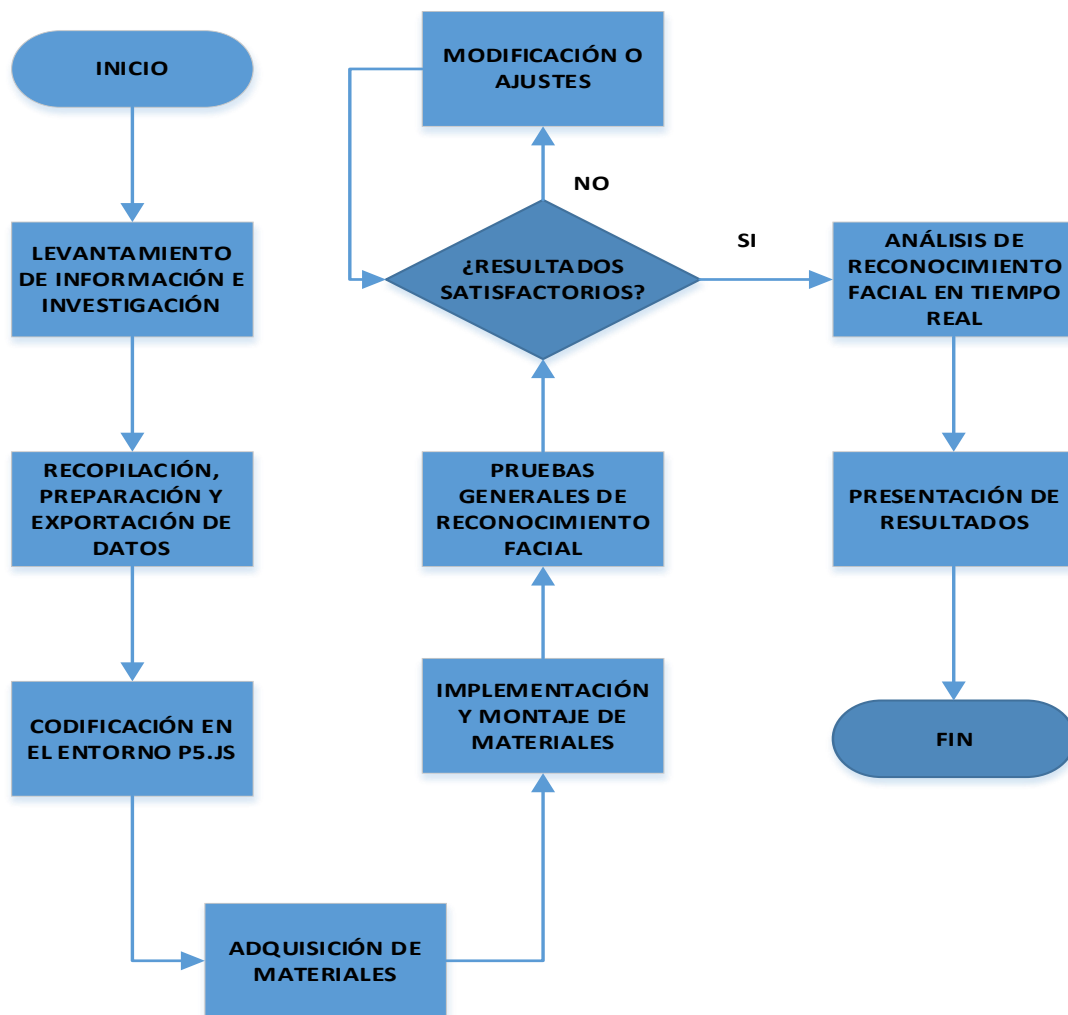


Figura 8. Esquema general del proceso de implementación del sistema.

De la misma manera se realizó un estudio bibliográfico para el desarrollo del análisis teórico del mismo, además de la parte práctica con sus respectivas pruebas y la obtención de resultados cualitativos y cuantitativos para proyectar una solución al problema planteado que consiste en el reconocimiento facial, Figura 8.

5.1 Diseño esquemático del proyecto

En la figura 9, se observa el diseño del proceso, el cual comienza con el usuario frente a la cámara, luego el programa realiza la comparación con la base de datos y envía la información al Arduino, el cual, especificará al relé a que usuario se le da acceso o no.

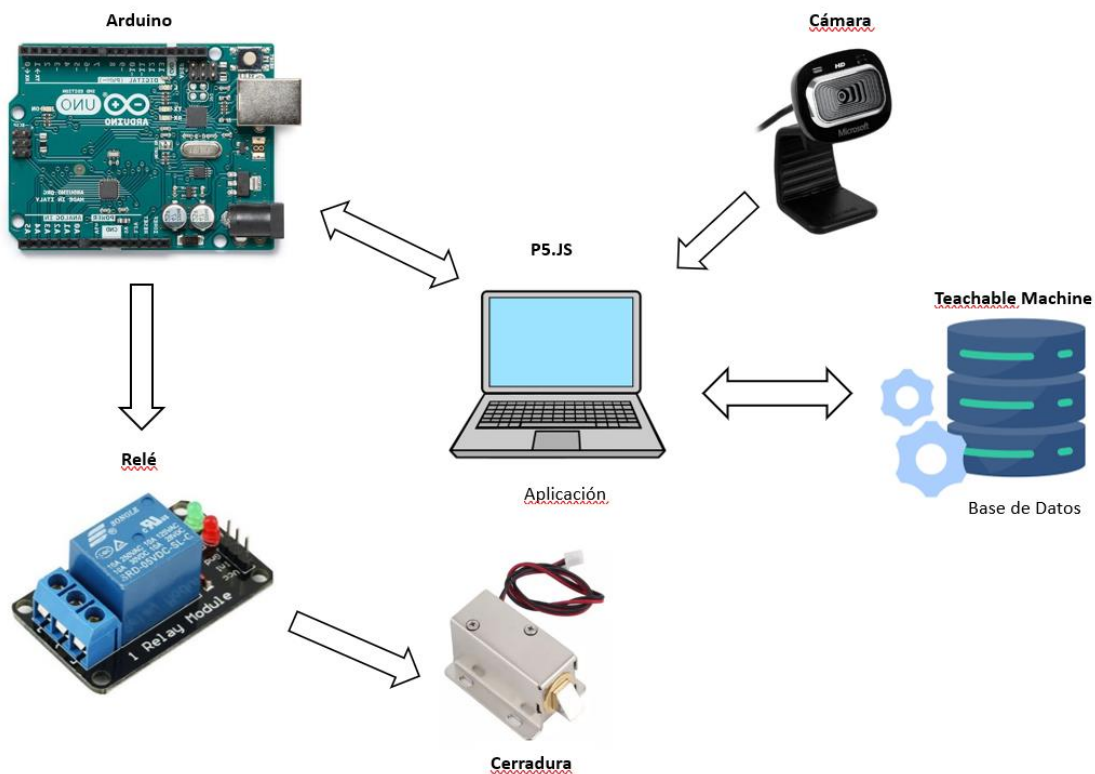


Figura 9. Diseño esquemático.

5.2 Diseño del modelo de clasificación de imágenes en Teachable Machine

Se crea un modelado de imagen en Teachable Machine, el cual está integrado por clases, que serían los perfiles que se creó para usuarios a los que se le autorice el acceso a la habitación.

5.2.1 Selección del proyecto deseado en Teachable Machine.

Teachable Machine cuenta con tres secciones distintas en las cuales realizar proyectos, estas secciones se dividen en:

- Proyecto de imagen.
- Proyecto de audio.
- Proyecto de posturas.

Para este estudio se utiliza la sección proyecto de imagen, Figura 10.

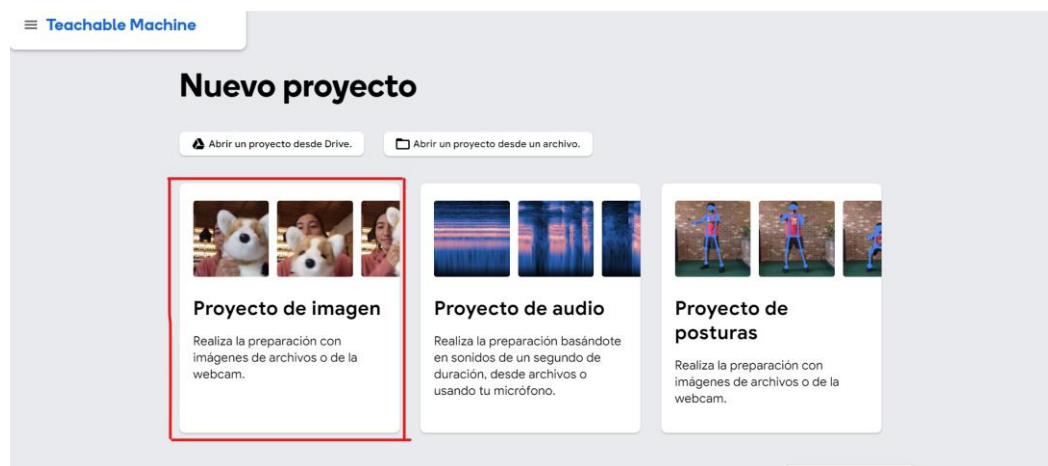


Figura 10. Selección de un nuevo proyecto Teachable Machine (Google, Teachable Machine, 2017).

Al elegir la sección proyecto de imagen, se muestran dos opciones a escoger, se selecciona el Modelo de imagen estándar, Figura 11.

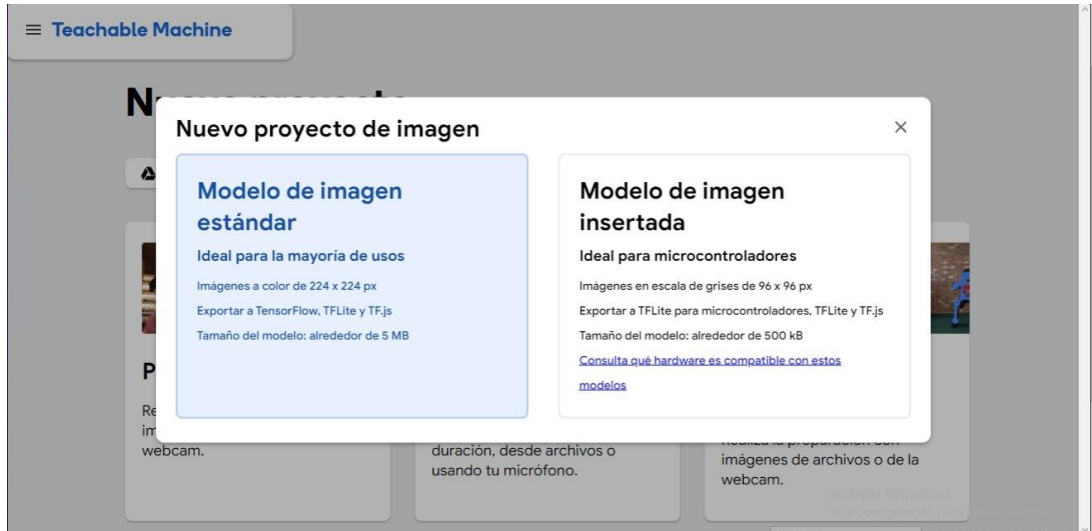


Figura 11. Selección del modelo a escoger para la clasificación de imagen (Google, Teachable Machine, 2017).

Se muestra el procedimiento a efectuar para lograr la clasificación de imágenes mediante aprendizaje automático:

5.2.1.1 Recopilación

Para empezar, se recopila y agrupa los ejemplos que el usuario desea que aprenda el ordenador dividiéndolos por clases o categorías, Figura 12.

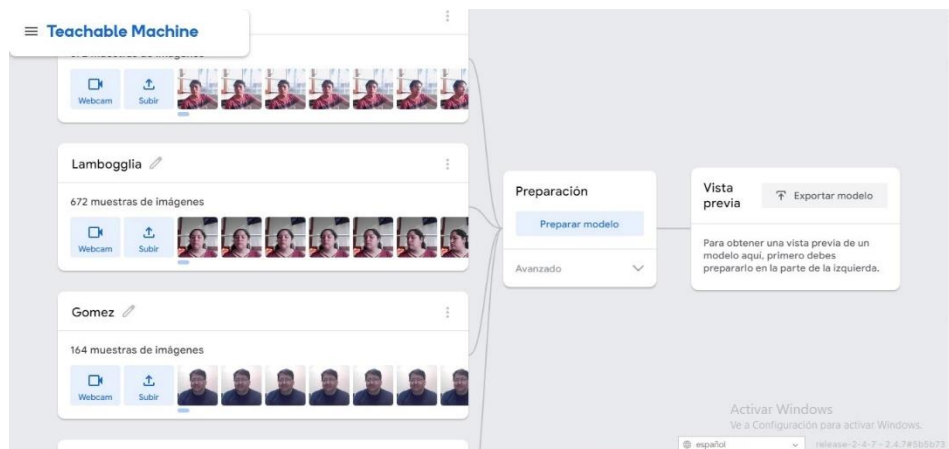


Figura 12. Creación de las clases a usar.

5.2.1.2 Preparación

Se prepara el modelo y se prueba al momento, para observar si puede clasificar correctamente los nuevos ejemplos, Figura 13.

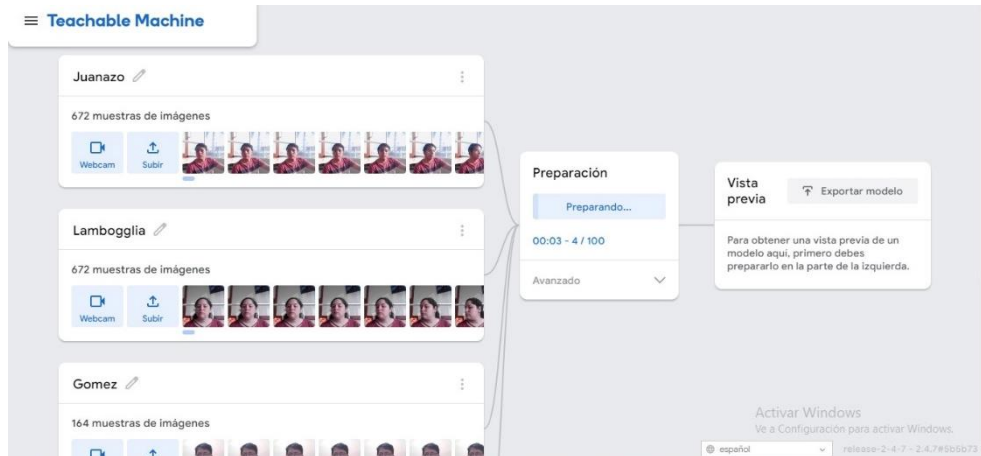


Figura 13. Preparación del modelo.

Una vez terminado el proceso de preparación, en el apartado vista previa, se muestra como la inteligencia artificial realiza la comparación en tiempo real de las clases creadas, mostrando el porcentaje de precisión del usuario frente a la cámara con la base de datos, Figura 27.

5.2.1.3 Exportación

Se exporta el modelo para proyectos, el cual se usa en: sitios web, aplicaciones, etc. Se puede descargar el modelo o alojarlo online, Figura 14.

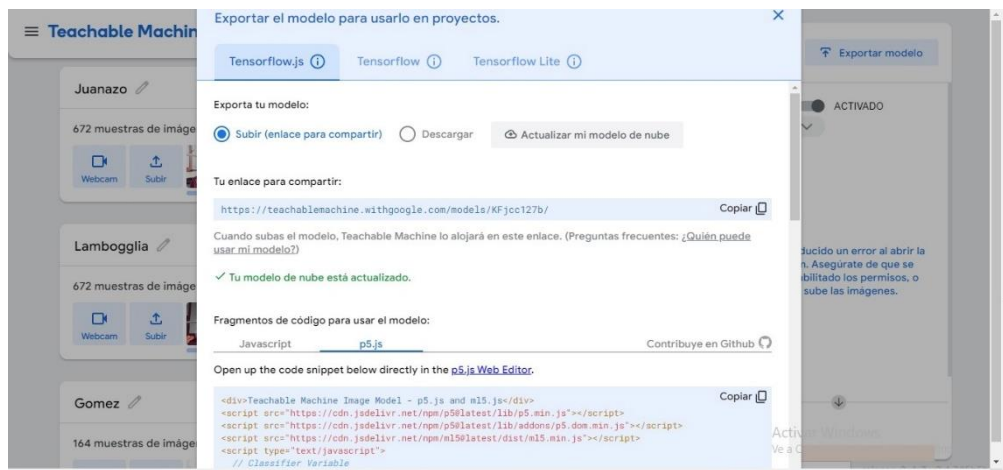


Figura 14. Sección para escoger la forma de exportar el modelo.

5.2.2 *Diseño del algoritmo en el entorno P5.js.*

A continuación, se realiza la codificación en la cual se integra el modelo exportado de Teachable Machine, para que al momento de que el rostro de la persona este frente a la cámara, se realice una comparación en las clases registradas del modelo, este código se realizó en el apartado sketch.js que es el centro de codificación, en el cual se efectúa cualquier modificación con respecto a la correcta ejecución del sistema de reconocimiento facial.

Aquí se muestran las variables a usar en la programación, como es el serial para la librería del puerto y el portname que es el puerto del Arduino, al igual que el link que se obtiene del modelo exportado de Teachable Machine, y la función de precarga para la captura del video junto con la librería ml5 para realizar la clasificación del modelo exportado, Figura 15.

```
1  const myImageModelURL =  
    'https://teachablemachine.withgoogle.com/models/KFjcc127b/';  
2  let myImageModel;  
3  let resultDiv;  
4  let serial;// variable to hold an instance of the serialport library  
5  let portName = 'COM3';// fill in your serial port name here  
6  let outByte = 0;// for outgoing data  
7  let video;  
8  let usuario;  
9  
10 function preload() {  
11   video = createCapture(VIDEO);  
12   myImageModel = ml5.imageClassifier(myImageModelURL+ 'model.json');  
13 }
```

Figura 15. Variables a usar y función de precarga para el video de la cámara.

Acto seguido, se muestra la función de setup, en el cual, se especifica el tamaño de la captura de video que se quiere mostrar en la preview de la aplicación P5.JS, también se realiza una nueva instancia para la librería del serialport y la abertura del puerto del Arduino a usar, Figura 16.

```
15 function setup() {
16   video.size(500,350)
17   createCanvas(50,50);
18   resultDiv = createElement('h1', '...');
19   serial = new p5.SerialPort(); // make a new instance of the
serialport library
20   serial.on('error', serialError); // callback for errors
21   serial.open(portName); // open a serial port
22   myImageModel.classify(video, gotResults);
23 }
```

Figura 16. Función setup de P5.JS.

Se muestra la función de obtener los resultados, en esta parte es donde se especifica los nombres usados en las clases del modelo de imagen, junto con los bytes de salida que se le asigna a cada uno, Figura 17 y 18.

```
29 function gotResults(err, results) {
30   if (err) console.log(err);
31   if (results) {
32     //console.log(results);
33     // Wait for 0.5 second before classifying again
34     setTimeout(() => myImageModel.classify(video, gotResults), 500);
35     if (results[0].confidence < 0.7) return;
36     resultDiv.html('Usuario: ' + results[0].label );
37   }
38
39   usuario=results[0].label;
40
41   if (usuario == 'Juanazo') {
42     //resultDiv.html('Usuario: ' + results[0].label);
43     //resultDiv.html('ACCESO CONCEDIDO');
44     outByte = 1;
45   } else if (usuario == 'Lamboggia') {
46     outByte = 2;
47   } else if (usuario == 'Yar') {
48     outByte = 3;
49   } else if (usuario == 'Gomez') {
50     outByte = 4;
51   } else if (usuario == 'Bautin') {
52     outByte = 5;
```

Figura 17. Función creada para la obtención de resultados.

```

53▼    } else if (usuario == 'Arriaga') {
54        outByte = 6;
55▼    } else if (usuario == 'Asqui') {
56        outByte = 7;
57▼    } else if (usuario == 'Baque') {
58        outByte = 8;
59▼    } else if (usuario == 'Rosero') {
60        outByte = 9;
61▼    } else if (usuario == 'Cardenas') {
62        outByte = 10;
63▼    } else if (usuario == 'Liz') {
64        outByte = 11;
65▼    } else if (usuario == 'Stefano') {
66        outByte = 12;
67▼    } else if (usuario == 'Carriel') {
68        outByte = 13;
69▼    } else if (usuario == 'Mendoza') {
70        outByte = 14;
71▼    } else if (usuario == 'Vacio') {
72        outByte = 0;
73    }

```

Figura 18. Nombres de usuarios.

En la siguiente sección de código se obtiene la visualización del nombre del usuario y los datos en la ventana de datos del serial, figura 19.

```

74
75    // send it out the serial port:
76    console.log('outByte: ', outByte)
77    serial.write(outByte);
78
79    console.log('caracteres: ', usuario)
80    serial.write(usuario);
81 }

```

Figura 19. Visualización de usuarios y datos en la consola.

También se coloca en el apartado index.html el título que se usa para el proyecto, las librerías a utilizar y una pequeña descripción del funcionamiento, Figura 20.



```
1
2 <html><head>
3 <title> Sistema de reconocimiento facial mediante P5.JS y Teachable Machine para la
4 apertura de una puerta</title>
5 <script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/p5.min.js">
6 </script>
7
8
9 <script language="javascript" type="text/javascript"
10 src="https://cdn.jsdelivr.net/npm/p5.serialserver@0.0.28/lib/p5.serialport.js">
11 </script>
12
13 <script
14 src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/addons/p5.dom.min.js"></script>
15 <script src="https://unpkg.com/ml5@latest/dist/ml5.min.js" type="text/javascript">
16 </script>
17 </head>
18
19 <body>
20 <h1>Reconocimiento Facial</h1>
21 <p>En este proyecto se implementó un modelo de clasificacion de imagenes con
22 Teachable Machine de Google. Si en la camara se muestra el rostro de la persona y esta
23 se compara con las clases creadas, enviará la información al Arduino, para la apertura
24 de la puerta.</p>
25 <script src="sketch.js"></script>
26 </body>
27 </html>
```

Figura 20. Sección de librerías y descripción de funcionamiento.

5.2.2.1 Aplicación para conectar el P5.js con el Arduino.

A través de la aplicación de p5.serialcontrol se realiza la conexión del algoritmo diseñado en la plataforma P5.js al Arduino que se vaya a utilizar, una vez dentro de la aplicación se procede a presionar el botón Rescan Ports para que identifique el puerto del Arduino cuando esté conectado al computador, cuando ya la aplicación identificó el puerto se presiona el botón open y luego se activa el apartado que dice console enabled, Figura 21.

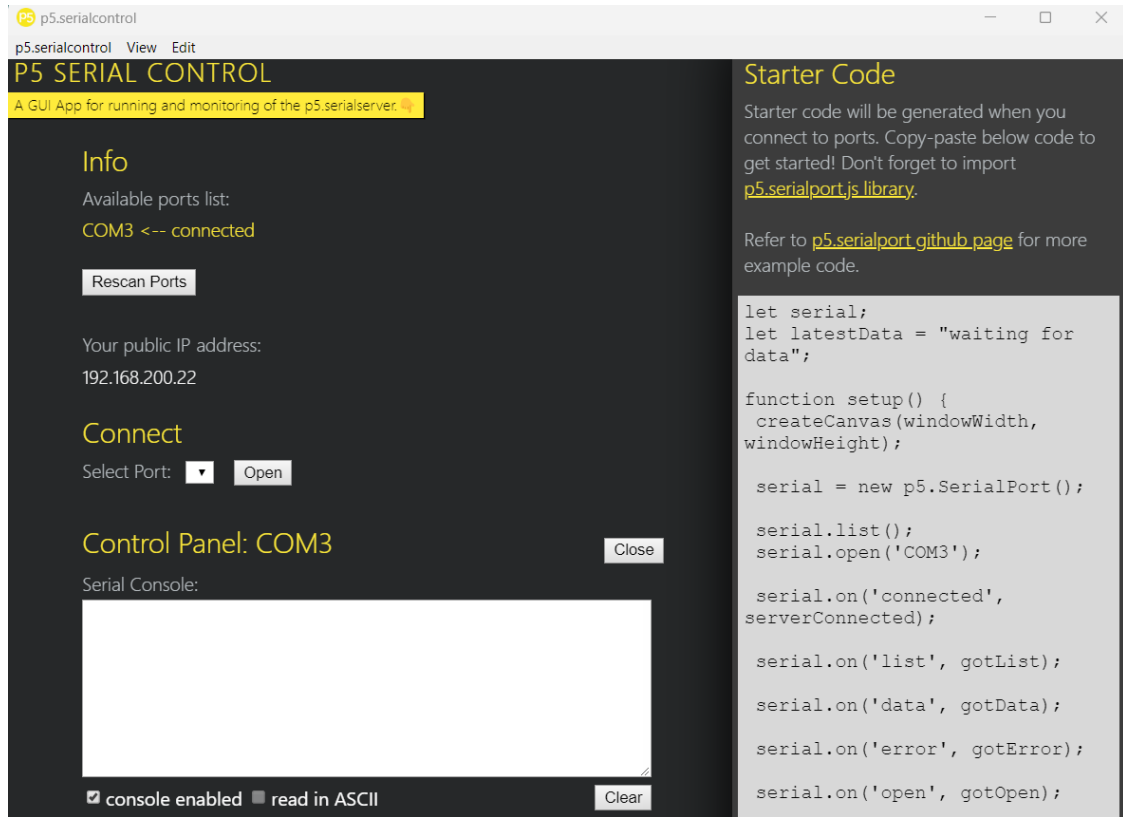


Figura 21. Aplicación p5.serialcontrol.

5.2.2.2 Codificación para el control en Arduino

El Arduino efectúa la acción que se haya asignado, en el caso de que se reconozca a las personas a las que se les da el acceso, el Arduino activa un led dependiendo de la persona a la que reconozca y hace el movimiento de abrir la cerradura.

Se crea variables para los diodos led y el relé para la cerradura, y en la función setup se establecen los pines como salida y el encendido del relé, Figura 22.


```

//
int ledPin1=2;
int ledPin2=3;
int ledPin3=4;
int relay=5;

void setup() {
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(ledPin3, OUTPUT);
  pinMode(relay, OUTPUT);
  Serial.begin(9600);
  digitalWrite(relay, HIGH);
}

```

Figura 22. Variables a usar y función setup.

A continuación, se describe la programación en la función loop, el proceso que el Arduino cumple, en el cual dependiendo el dato que esté entrando realiza una condición u otra, en este caso a cada condición se le ha asignado que se encienda un diodo led correspondiente y la apertura o el bloqueo de la cerradura, figura 23 y 24.

```

void loop() {
  if (Serial.available()>0){
    int inByte = Serial.read();
    Serial.print(inByte);
    delay(1);
    if(inByte==1){
      digitalWrite(ledPin1, HIGH);
      digitalWrite(ledPin2, LOW);
      digitalWrite(ledPin3, LOW);
      digitalWrite(relay, LOW); //HIGH es cerrar y LOW es abrir
      //Serial.print("juanazo");
    }else if(inByte==2){
      digitalWrite(ledPin2, HIGH);
      digitalWrite(ledPin1, LOW);
      digitalWrite(ledPin3, LOW);
      digitalWrite(relay, HIGH); //HIGH es cerrar y LOW es abrir
      //Serial.print("lambogglia");
    }else if(inByte==3){
      digitalWrite(ledPin2, HIGH);
      digitalWrite(ledPin1, LOW);
      digitalWrite(ledPin3, LOW);
      digitalWrite(relay, HIGH); //HIGH es cerrar y LOW es abrir
      //Serial.print("Yar");
    }
  }
}

```

Figura 23. Void loop, asignación de pines, etapa 1.

```

}else if(inByte==4){
  digitalWrite(ledPin1,HIGH);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("gomez");
}else if(inByte==5){
  digitalWrite(ledPin1,HIGH);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Bautin");
}else if(inByte==6){
  digitalWrite(ledPin2,HIGH);
  digitalWrite(ledPin1,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Arriaga");
}else if(inByte==7){
  digitalWrite(ledPin2,HIGH);
  digitalWrite(ledPin1,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Asqui");
}else if(inByte==8){
  digitalWrite(ledPin1,HIGH);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Baque");

```

Figura 24. Void loop, asignación de pines, etapa 2.

Entonces al byte de entrada 1 se le asignó la apertura de la cerradura mientras al resto se le niega el acceso a la puerta, lo mismo para la condición de Vacío, es decir el dato es 0, en este se enciende el diodo led correspondiente y se cierra la cerradura, figura 25 y 26.

```

}else if(inByte==9){
  digitalWrite(ledPin1,HIGH);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Rosero");
}else if(inByte==10){
  digitalWrite(ledPin2,HIGH);
  digitalWrite(ledPin1,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Cardenas");
}else if(inByte==11){
  digitalWrite(ledPin2,HIGH);
  digitalWrite(ledPin1,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Liz");
}else if(inByte==12){
  digitalWrite(ledPin1,HIGH);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Stefano");
}else if(inByte==13){
  digitalWrite(ledPin2,HIGH);
  digitalWrite(ledPin1,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Carriel");

```

Figura 25. Void loop, asignación de pines, etapa 3.

```
}else if(inByte==14){
  digitalWrite(ledPin1,HIGH);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Mendoza");
} else if(inByte==0) {
  digitalWrite(ledPin1,LOW);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,HIGH);
  digitalWrite(relay,HIGH);
  //Serial.print("apagado");
}
} //delay(5);
}
```

Figura 26. Void loop, asignación de pines, etapa 4.

VI RESULTADOS

Una vez concluida la construcción se realizaron varias pruebas para la evaluación del sistema de reconocimiento facial en tiempo real, la cual permitió realizar ajustes y mejoras en su optimización.

6.1 Modelo de imagen como base de datos en Teachable Machine

Para la comprobación de las clases del modelo de imagen, el software Teachable Machine cuenta con un apartado en el cual se muestra la cámara en vivo, y de esa forma corroborar la comparación entre la información almacenada y lo que se observa en cámara, Figura 27.

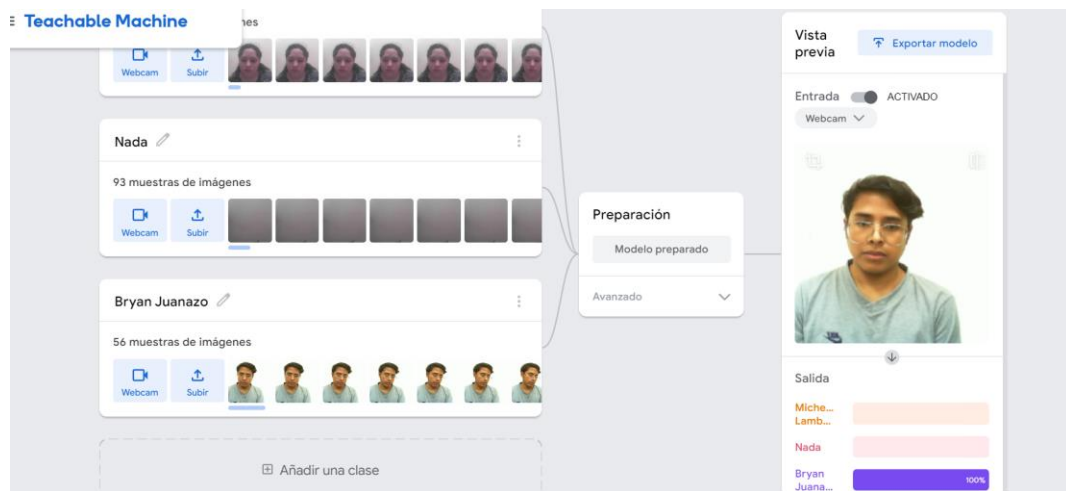


Figura 27. Comprobación de datos en Teachable Machine.

En las primeras pruebas de la base de datos, se utilizó un nombre y un apellido de cada integrante y también se asignó una clase llamada nada. Una vez lista las clases, se preparó el modelo con las especificaciones por default en el apartado de avanzado, épocas de 50, tamaño del lote de 16 y una tasa de aprendizaje del 0,001, Figura 28.

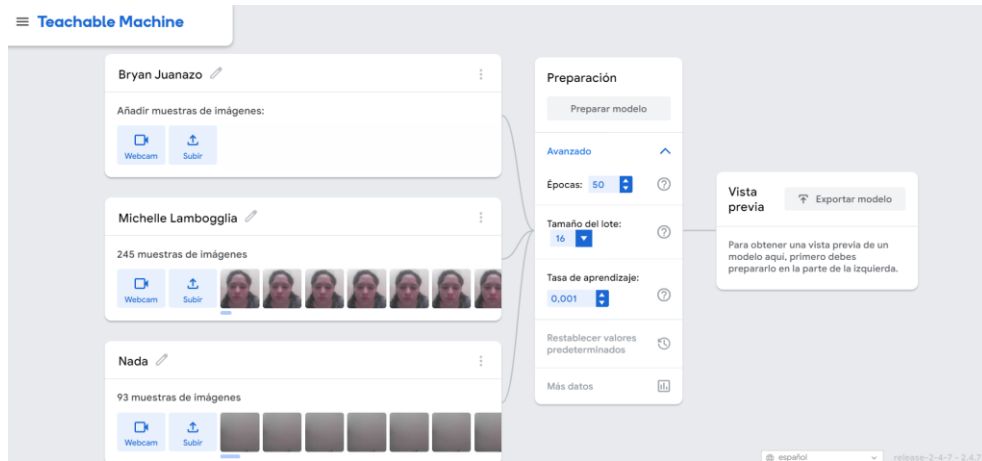


Figura 28. Registro de datos.

Al final, se incluyeron 15 clases en la base de datos, en la que, 14 corresponden a los usuarios y uno llamado vacío. En el caso de los usuarios, se decidió nombrarlos solo por el apellido o solo por el nombre, esto porque en las pruebas realizadas en el P5.JS, en la que se usó un nombre y un apellido, no generaba el outbyte correspondiente de cada usuario, por lo que al Arduino le llegaba el outbyte de 0.

Por último, en la preparación del modelo solo se realizó un cambio en el tamaño de las épocas de 50 a 100, Figura 29.

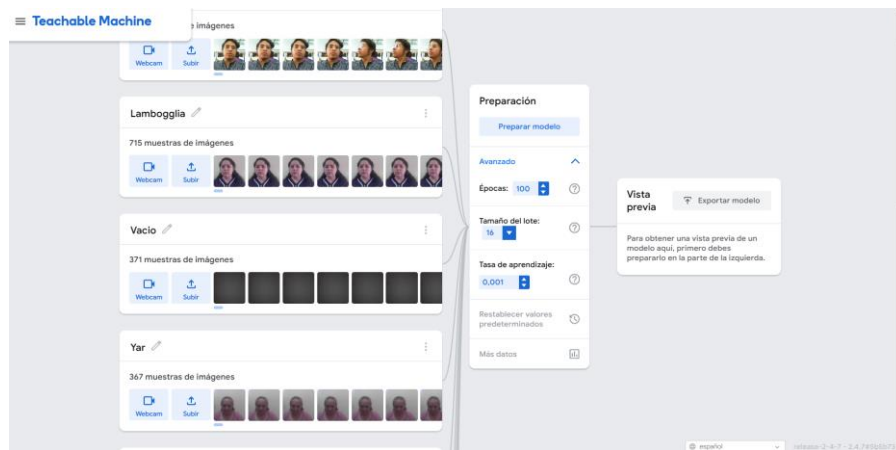


Figura 29. Preparación del modelo.

6.2 Algoritmo en P5.JS y Arduino.

En el algoritmo de P5.JS sin la conexión del Arduino, se realizó pruebas con la base de datos donde se usó el nombre y apellido de los usuarios y la clase nada, durante este proceso, se observó que se presentaban errores, ya que, no se obtenía el dato de salida asignado a cada usuario.

Por lo que se decidió, colocar solo los apellidos o solo los nombres en los integrantes y reemplazar la palabra nada por vacío, de esa forma se obtuvo los datos correspondientes, Figura 30.

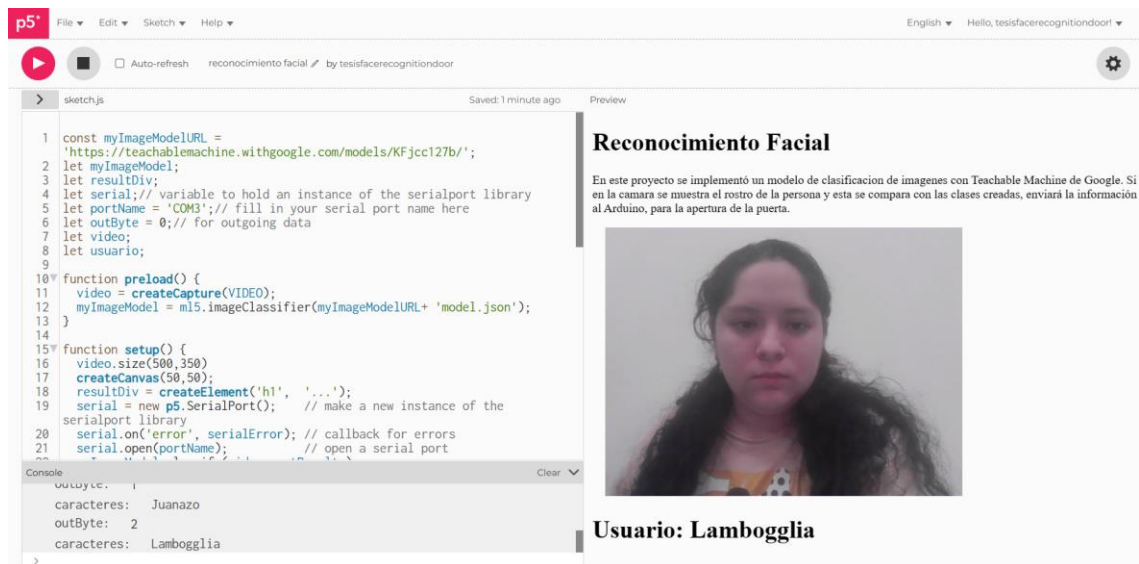


Figura 30. Prueba de algoritmo en P5.JS.

Las primeras pruebas de este algoritmo, fueron realizadas con conexiones en el protoboard, para comprobar el funcionamiento de la cerradura y de los diodos led, Figura 31.

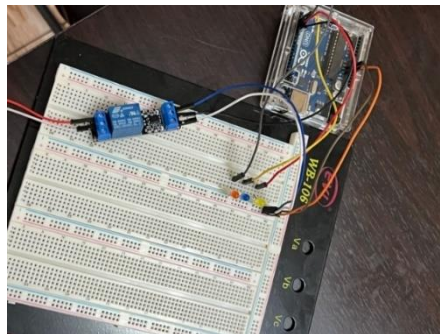


Figura 31. Conexión en protoboard.

6.3 Prueba y Error en la etapa de evaluación del sistema

Durante el proceso de prueba se presentaron algunas complicaciones, como que el código en p5.js botaba un error de librerías en el apartado index.html por lo que se procedió a investigar las librerías correctas para lograr que el código funcione con normalidad, otro error que se presentó fue que los diodos leds se mantenían encendidos no importara que usuario se encontraba frente a la cámara y la cerradura eléctrica se abría y cerraba, Analizando posibles soluciones se logró llegar al resultado deseado, Figura 32.

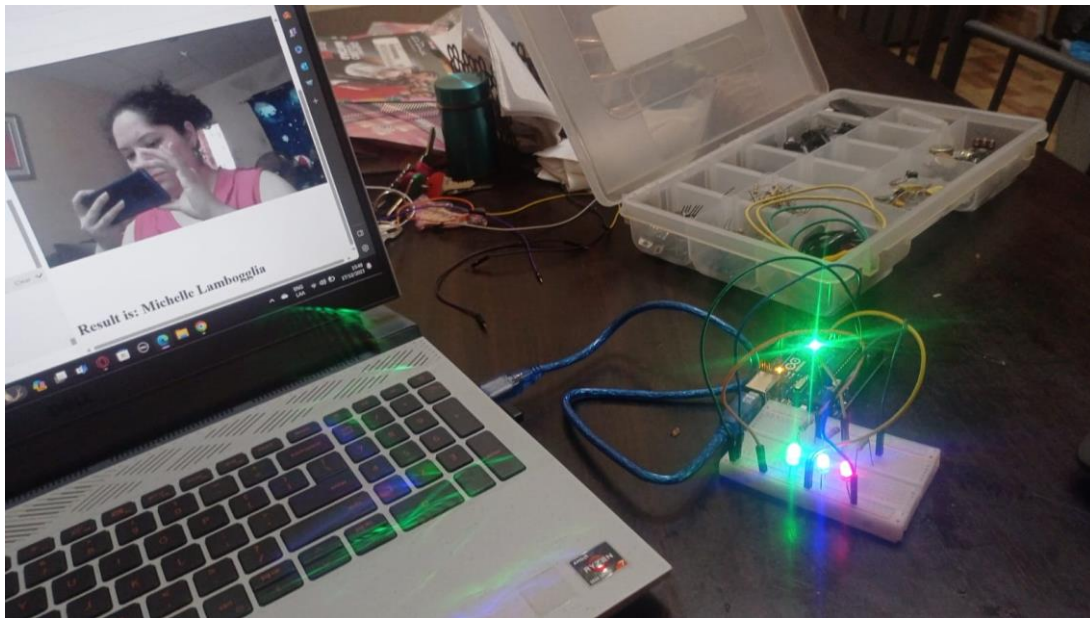


Figura 32. Prueba y Error.

6.4 Prueba final del sistema

Se preparó la maqueta, además de realizar pruebas en ella. Para las primeras comprobaciones se programó, que se abriera a uno de los usuarios, mientras que al otro se le permanezca cerrada, Figura 33.



Figura 33. Pruebas en la maqueta.

Para la prueba final, se decidió que, para cada clase registrada en la base de datos, se asignará un diodo led, y en el proceso de funcionamiento, un usuario tendrá acceso mientras que el otro no. Es decir, en el momento que la persona coloque su rostro frente a la cámara, abrirá la cerradura, y en el momento en que el usuario traspasa la puerta, la cámara detectará el dato de vacío, por lo que cerrará la cerradura y activará un diodo led.

Como se muestra en la figura 34, se realizó la prueba en maqueta ya con todo conectado, la cerradura colocada en la puerta y el Arduino conectado a la computadora para comprobar el correcto funcionamiento del proyecto.



Figura 34. Pruebas para apertura de un usuario.

Finalmente, en la figura 35, se observa la parte frontal de la maqueta en la que el circuito está etiquetado con su respectivo nombre, del mismo modo, led verde para usuario 1, led azul para el usuario 2, led rojo para Cerrado, la cámara ubicada en la esquina superior de la maqueta y el relé en la parte inferior.

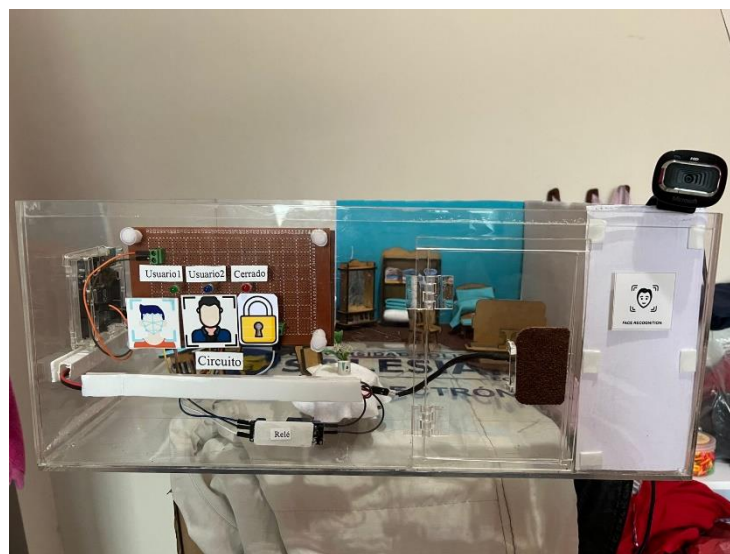


Figura 35. Visualización de la maqueta final.

En la figura 36 y 37 se observa, el acabo final de la maqueta, tanto en su interior como exterior, en la cual se plasmó la habitación de un hogar.



Figura 36. Parte interna de la maqueta.



Figura 37. Visualización de maqueta completa.

6.5 Análisis de datos según la matriz de confusiones de Teachable Machine

Aquí se resume el grado de precisión de las predicciones del modelo creado a través de la obtención de datos generados por el software Teachable Machine. Esta matriz se utiliza para determinar que clases resultan confusas para el modelo. En el eje Y (clase) representa la clase de las muestras. El eje X (predicción) representa la clase a la que el modelo asigna las muestras después del aprendizaje. Por lo tanto, si la clase de una muestra es “Lamboggliá”, pero su predicción es “Juanazo”, quiere decir que, después de aprenderse tus datos, el modelo determina incorrectamente que una muestra de “Lamboggliá”, pertenece a “Juanazo”.

Por lo general esto significa que las dos clases comparten características que el modelo detecta y que una muestra concreta de “Lamboggliá” se parece más a las muestras de “Juanazo”,

Figura 38.

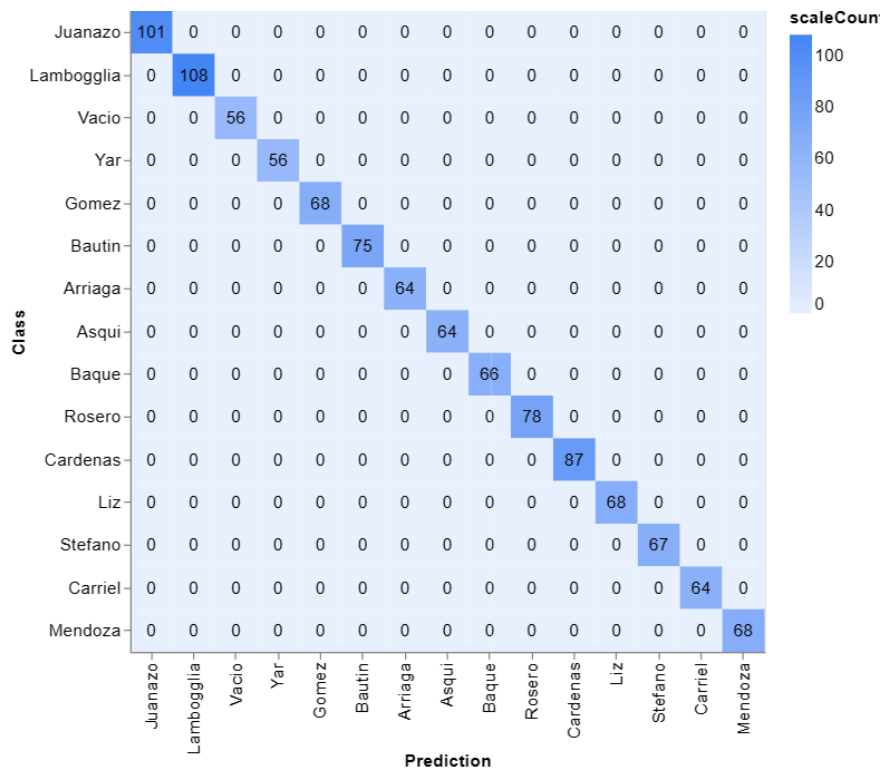


Figura 38. Matriz de confusión generada

VII CRONOGRAMA

El cronograma fue realizado en una plantilla de diagrama de Gantt, en el eje vertical se encuentran las actividades realizadas y en el eje horizontal se visualiza los días y el mes del calendario del proyecto, las barras que se muestran en el diagrama, representan los días en que empezó y terminó la actividad.

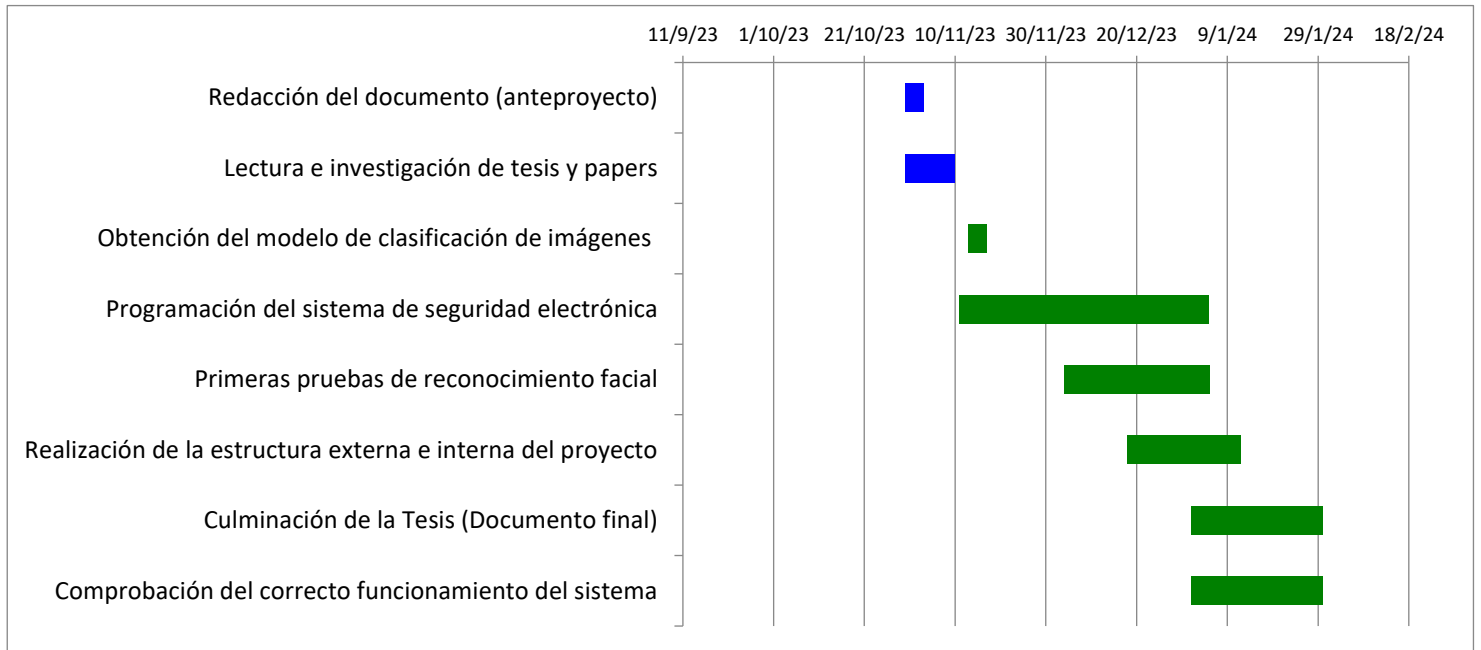


Figura 39. Diagrama de Gantt del cronograma del proyecto

En la tabla 1, se muestra el cronograma de actividades, el cual se divide en: las tareas que son las actividades realizadas, los responsables en este caso los autores, fecha de inicio en donde se especifica mes/día, fecha final igual al anterior, la sección días que contabiliza el tiempo transcurrido entre la fecha de inicio y la fecha final y por último la sección estado en donde se muestra el estatus de las actividades.

Tabla 1.

Actividades del proyecto con Gantt.

TAREAS	RESPONSABLE	FECHA DE INICIO	FECHA FINAL	DÍAS	ESTADO
Redacción del documento (anteproyecto)	Lamboggia M. - Juanazo A.	10/30	11/3	4	Completado
Lectura e investigación de tesis y papers	Lamboggia M. - Juanazo A.	10/30	11/10	11	Completado
Obtención del modelo de clasificación de imágenes	Lamboggia M. - Juanazo A.	11/13	11/17	4	Completado
Programación del sistema de seguridad electrónica	Lamboggia M. - Juanazo A.	11/11	1/5	55	Completado
Primeras pruebas de reconocimiento facial	Lamboggia M. - Juanazo A.	12/4	1/5	32	Completado
Realización de la estructura externa e interna del proyecto	Lamboggia M. - Juanazo A.	12/18	1/12	25	Completado
Culminación de la tesis (documento final)	Lamboggia M. - Juanazo A.	1/1	1/30	29	Completado
Comprobación del correcto funcionamiento del sistema	Lamboggia M. - Juanazo A.	1/1	1/30	29	Completado

VIII PRESUPUESTO

En la tabla 2 se observará los costos de los componentes a utilizar para la realización del proyecto. Estos valores varían de acuerdo al lugar de compra, serán efectuados en su costo por los autores.

Tabla 2.

Costos de componentes.

Componentes	Costo
Arduino Uno	\$22
Cerradura eléctrica	\$43
Cámara de PC con conexión USB	\$26
Total	\$91.00

IX CONCLUSIONES

La base de datos realizada en este proyecto, fue de vital importancia, ya que permitió obtener el modelado de imagen basado en inteligencia artificial, realizando la comparación exitosa entre los usuarios de la base de datos.

Con los resultados obtenidos en P5.JS, se puede deducir que las librerías en el apartado index.html deben ser las indicadas, porque así el código este correcto, si las librerías son erróneas, no se obtendrá ningún resultado al ejecutar el programa y se mostrará error de librerías en el apartado console.

En el Arduino, se identificó un “delay”, que hacía que la cerradura se tardara 10 segundos en reaccionar. Una vez detectado el problema, se pudo corregir y lograr que la cerradura se abra y se cierre al instante.

Con las pruebas realizadas se concluye que el proyecto, satisface los objetivos planteados, ya que el prototipo realiza la comparación en tiempo real de la base de datos con el usuario que se muestra ante la cámara, dando así pie a que el programa P5.JS envíe información de salida al Arduino, para activar o mantener desactivado la cerradura.

X RECOMENDACIONES

En este Proyecto se utilizó la clasificación de modelo de imagen de la plataforma Teachable Machine, en la cual, además de insertar imágenes a través de la cámara, también se puede agregar archivos de imágenes almacenadas que se tenga en la computadora, también se recomienda usar Open CV para el aprendizaje automático.

Al momento de exportar el modelo creado en la plataforma de Teachable Machine, este ofrece fragmentos de codificación a usar tanto en P5.JS como en JavaScript, de acuerdo al tipo de modelo creado. La plataforma P5.JS, al igual que Teachable Machine, ofrece ejemplos de diferentes códigos, los cuales pueden ayudar para la creación de un proyecto.

Para mayor seguridad, se recomienda usar dos filtros para la apertura de una puerta principal de una vivienda, los cuales son: reconocimiento facial y huella dactilar. Esto debido a que Teachable Machine, no es tan preciso al momento de reconocer a un individuo que no se encuentra en la base de datos.

Se puede reemplazar la laptop por una Raspberry Pi B+ el cual es compatible con Teachable Machine para el aprendizaje automático. Lo esencial para trabajar con un Raspberry Pi es tener una micro sd, en el cual se instala el Raspbian, que es el sistema operativo de este, una cámara Raspberry Pi y con eso puedes empezar a trabajar.

XI REFERENCIAS

- Aldundia. (11 de febrero de 2021). *Bizkaia*. Obtenido de https://www.bizkaia.eus/home2/archivos/DPTO3/Temas/Inspecci%C3%B3n%20y%20control/VC_INTIMIDAD%20documento%20VC%20firmado.pdf?hash=a612de414e4b5f70f92a5a128f389818&idioma=CA
- Algotive. (2022). ¿Cómo funciona la inteligencia artificial en seguridad?
- Ashwini, Balaji, V., Srinivasan, S., & Monisha, K. (2018). *Real Time Facial Recognition System*. Coimbatore: Springer.
- Blanco, F. R. (2020). Raspberry Pi, Machine Learning & Teachable Machine: Cómo detectar si lleva puesta la mascarilla con vision Artificial . *Un informático en el lado del mal*.
- Draz, U. (2022). What is p5.js? *Educative*.
- Duckerman, W. (2022). ¿Qué es la tecnología de reconocimiento facial de IA y cómo funciona? *BRITA Inteligencia Artificial*.
- España, G. d. (2023). Qué es la Inteligencia Artificial. *Plan de Recuperación, Transformación y Resiliencia*.
- Fernández, C. C. (14 de junio de 2019). *CDD Economy & Law*. Obtenido de <https://centrodesarrollodirectivo.es/impacto-avances-tecnologicos-privacidad/>
- Google. (2017). Teachable Machine.
- Google. (2019). Conceptos Básicos. *Teachable Machine*.
- Google. (2022). TensorFlow.js: Make your own "Teachable Machine" using transfer learning with TensorFlow.js. *Google Codelabs*.
- ID, E. (2022). ¿Cómo funciona el reconocimiento facial? Tecnología facial. *Electronic IDentification A Signicat Company*.

Lacort, J. (2017). La claves de los sistemas de reconocimiento facial:¿cuál es su verdadero nivel de seguridad? *Xataka*.

Maite, D. I. (2021). La intimidad. *Bizkaia*.

Mecatronica. (2024). UNO R3+Cable Usb Atmega328p AR. *Mecatronica Robotica y Electronica*.

NYU. (2020). Lab: Serial output from P5.js Using the p5.serialport Library. *Interactive*.

Sarthak. (2021). P5.js Introduction. *GeeksforGeeks*.

servicenow. (2024). ¿Qué es la IA para la seguridad? *servicenow*.

Services, T. (2023). OpenCV y Deep Learning - Visión por computador en redes neuronales. *Linkedin*.

Techzizou. (2022). Build a custom Image classification Android app using Teachable Machine (old version TF App). *Medium*.

UTEC. (2020). Teachable Machine - AI para todos. *aiUTEC Challenge*.

Zambetti, N. (2020). ¿Sabes qué es un Arduino y para qué sirve? *Aquae Fundación*.

XII ANEXO

Código del sketch.js en P5.JS

```
const myImageModelURL = 'https://teachablemachine.withgoogle.com/models/KFjcc127b/';
let myImageModel;
let resultDiv;
let serial;// variable to hold an instance of the serialport library
let portName = 'COM3';// fill in your serial port name here
let outByte = 0;// for outgoing data
let video;
let usuario;

function preload() {
  video = createCapture(VIDEO);
  myImageModel = ml5.imageClassifier(myImageModelURL+ 'model.json');
}

function setup() {
  video.size(500,350)
  createCanvas(50,50);
  resultDiv = createElement('h1', '...');
  serial = new p5.SerialPort(); // make a new instance of the serialport library
  serial.on('error', serialError); // callback for errors
  serial.open(portName); // open a serial port
  myImageModel.classify(video, gotResults);
}

function serialError(err) {
  console.log('Something went wrong with the serial port. ' + err);
}

function gotResults(err, results) {
  if (err) console.log(err);
  if (results) {
    //console.log(results);
    // Wait for 0.5 second before classifying again
    setTimeout(() => myImageModel.classify(video, gotResults), 500);
    if (results[0].confidence < 0.7) return;
    resultDiv.html('Usuario: ' + results[0].label );
  }

  usuario=results[0].label;

  if (usuario =='Juanazo') {
    //resultDiv.html('Usuario: ' + results[0].label);
    //resultDiv.html('ACCESO CONCEDIDO');
    outByte = 1;
  } else if (usuario =='Lamboggliia') {
    outByte = 2;
  } else if (usuario =='Yar') {
```

```
    outByte = 3;
  } else if (usuario == 'Gomez') {
    outByte = 4;
  } else if (usuario == 'Bautin') {
    outByte = 5;
  } else if (usuario == 'Arriaga') {
    outByte = 6;
  } else if (usuario == 'Asqui') {
    outByte = 7;
  } else if (usuario == 'Baque') {
    outByte = 8;
  } else if (usuario == 'Rosero') {
    outByte = 9;
  } else if (usuario == 'Cardenas') {
    outByte = 10;
  } else if (usuario == 'Liz') {
    outByte = 11;
  } else if (usuario == 'Stefano') {
    outByte = 12;
  } else if (usuario == 'Carriel') {
    outByte = 13;
  } else if (usuario == 'Mendoza') {
    outByte = 14;
  } else if (usuario == 'Vacio') {
    outByte = 0;
  }

  // send it out the serial port:
  console.log('outByte: ', outByte)
  serial.write(outByte);

  console.log('caracteres: ', usuario)
  serial.write(usuario);
}
```

Fuentes del index.html de P5.JS

```
<html><head>
  <title> Sistema de reconocimiento facial mediante P5.JS y Teachable Machine para
  la apertura de una puerta</title>

  <script
  src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/p5.min.js"></script>

  <script language="javascript" type="text/javascript"
  src="https://cdn.jsdelivr.net/npm/p5.serialserver@0.0.28/lib/p5.serialport.js"></scr
  ipt>

  <script
  src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/addons/p5.dom.min.js"></scri
  pt>
  <script src="https://unpkg.com/ml5@latest/dist/ml5.min.js"
  type="text/javascript"></script>
</head>

  <body>
    <h1>Reconocimiento Facial</h1>
    <p>En este proyecto se implementó un modelo de clasificacion de imagenes con
    Teachable Machine de Google. Si en la camara se muestra el rostro de la persona y
    esta se compara con las clases creadas, enviará la información al Arduino, para la
    apertura de la puerta.</p>
    <script src="sketch.js"></script>
  </body>
</html>
```

Código de Arduino

```
int ledPin1=2;
int ledPin2=3;
int ledPin3=4;
int relay=5;

void setup() {
  pinMode(ledPin1,OUTPUT);
  pinMode(ledPin2,OUTPUT);
  pinMode(ledPin3,OUTPUT);
  pinMode(relay,OUTPUT);
  Serial.begin(9600);
  digitalWrite(relay,HIGH);
}

void loop() {
  if (Serial.available()>0){
    int inByte = Serial.read();
    Serial.print(inByte);
    delay(1);
    if(inByte==1){
      digitalWrite(ledPin1,HIGH);
      digitalWrite(ledPin2,LOW);
      digitalWrite(ledPin3,LOW);
      digitalWrite(relay,LOW);//HIGH es cerrar y LOW es abrir
      //Serial.print("juanazo");
    }else if(inByte==2){
      digitalWrite(ledPin2,HIGH);
      digitalWrite(ledPin1,LOW);
      digitalWrite(ledPin3,LOW);
      digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
      //Serial.print("lamboggia");
    }else if(inByte==3){
      digitalWrite(ledPin2,HIGH);
      digitalWrite(ledPin1,LOW);
      digitalWrite(ledPin3,LOW);
      digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
      //Serial.print("Yar");
    }else if(inByte==4){
      digitalWrite(ledPin1,HIGH);
      digitalWrite(ledPin2,LOW);
      digitalWrite(ledPin3,LOW);
      digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
      //Serial.print("gomez");
    }else if(inByte==5){
      digitalWrite(ledPin1,HIGH);
      digitalWrite(ledPin2,LOW);
      digitalWrite(ledPin3,LOW);
      digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
      //Serial.print("Bautin");
    }
  }
}
```

```

}else if(inByte==6){
    digitalWrite(ledPin2,HIGH);
    digitalWrite(ledPin1,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Arriaga");
}else if(inByte==7){
    digitalWrite(ledPin2,HIGH);
    digitalWrite(ledPin1,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Asqui");
}else if(inByte==8){
    digitalWrite(ledPin1,HIGH);
    digitalWrite(ledPin2,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Baque");
} else if(inByte==0) {
    digitalWrite(ledPin1,LOW);
    digitalWrite(ledPin2,LOW);
    digitalWrite(ledPin3,HIGH);
    digitalWrite(relay,HIGH);
    //Serial.print("apagado");
}else if(inByte==9){
    digitalWrite(ledPin1,HIGH);
    digitalWrite(ledPin2,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Rosero");
}else if(inByte==10){
    digitalWrite(ledPin2,HIGH);
    digitalWrite(ledPin1,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Cardenas");
}else if(inByte==11){
    digitalWrite(ledPin2,HIGH);
    digitalWrite(ledPin1,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Liz");
}else if(inByte==12){
    digitalWrite(ledPin1,HIGH);
    digitalWrite(ledPin2,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Stefano");
}else if(inByte==13){
    digitalWrite(ledPin2,HIGH);
    digitalWrite(ledPin1,LOW);
    digitalWrite(ledPin3,LOW);
    digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
    //Serial.print("Carriel");

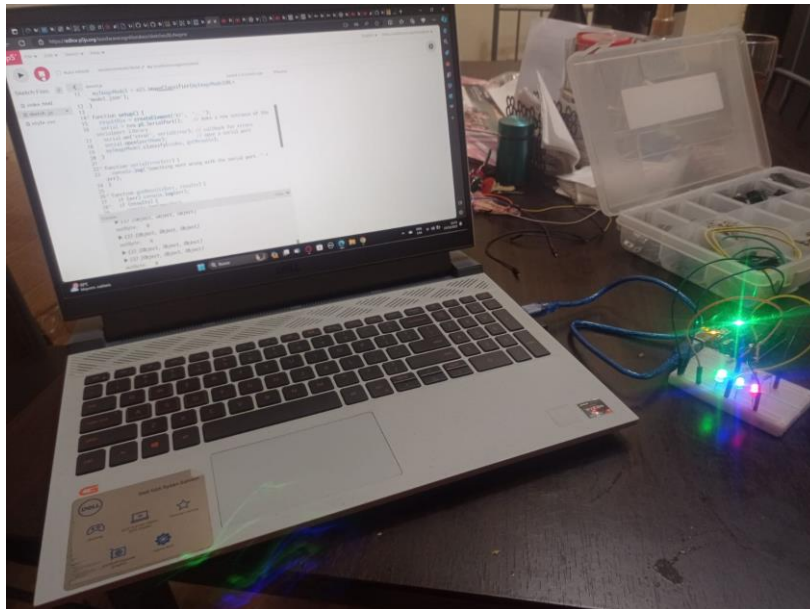
```

```
}else if(inByte==14){
  digitalWrite(ledPin1,HIGH);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,LOW);
  digitalWrite(relay,HIGH);//HIGH es cerrar y LOW es abrir
  //Serial.print("Mendoza");
} else if(inByte==0) {
  digitalWrite(ledPin1,LOW);
  digitalWrite(ledPin2,LOW);
  digitalWrite(ledPin3,HIGH);
  digitalWrite(relay,HIGH);
  //Serial.print("apagado");
}
} //delay(5);
}
```


Fotos de pruebas e implementación de maqueta



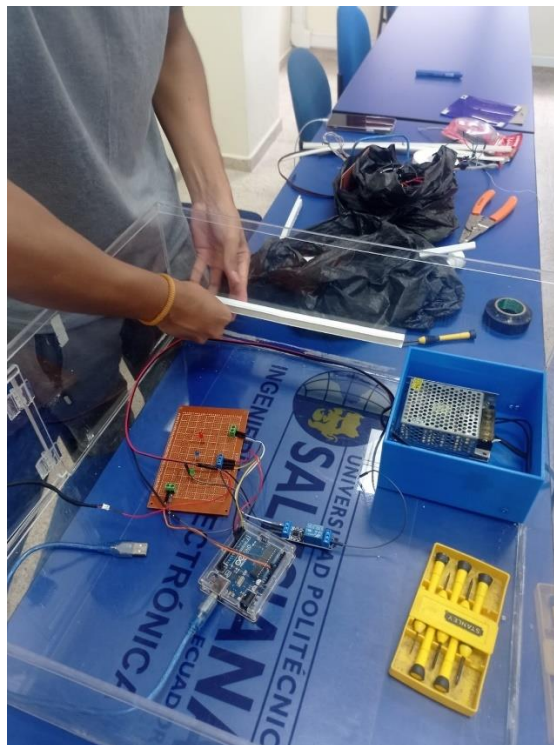
Pruebas de encendido de diodos led.



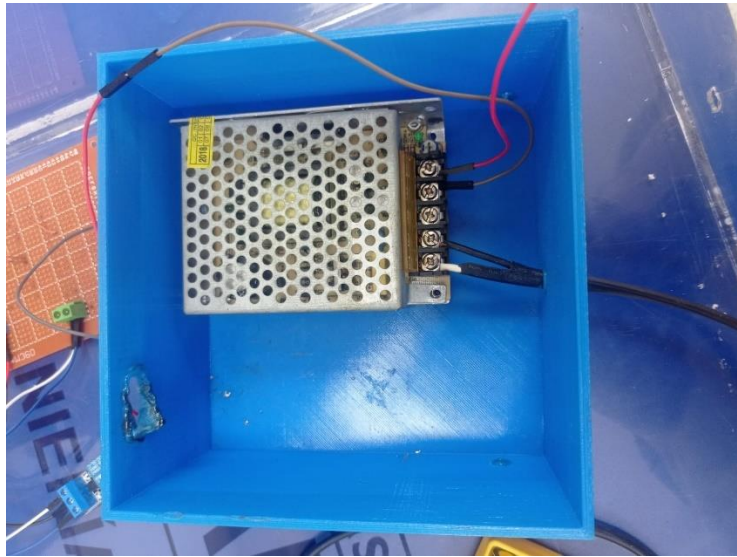
Primeros fallos del algoritmo.



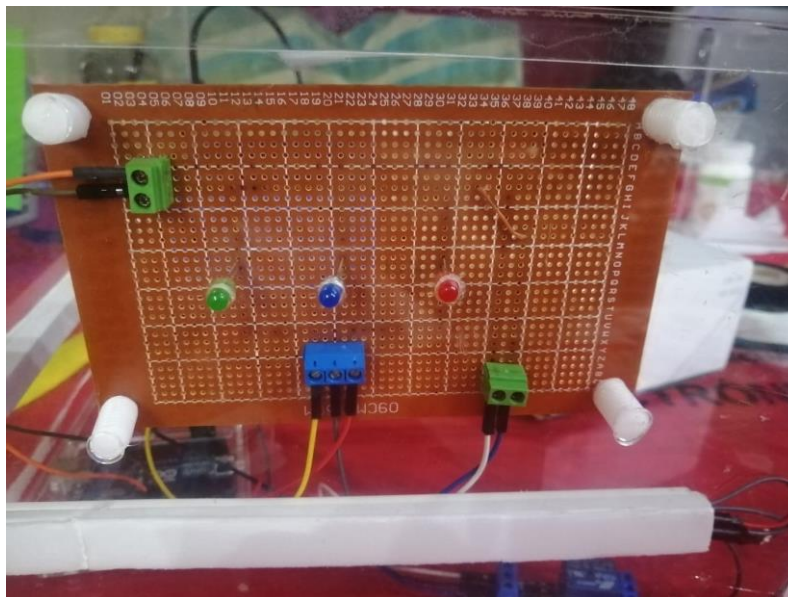
Pruebas de funcionamiento del algoritmo.



Implementación de elementos electrónicos para circuito.



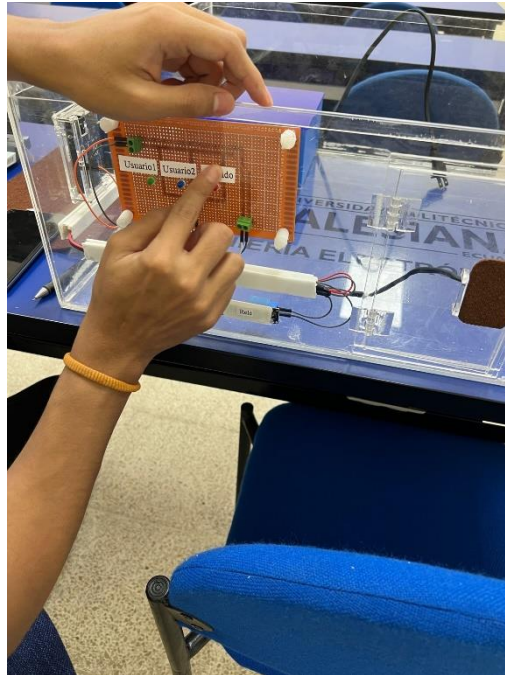
Fuente de voltaje de 12 voltios.



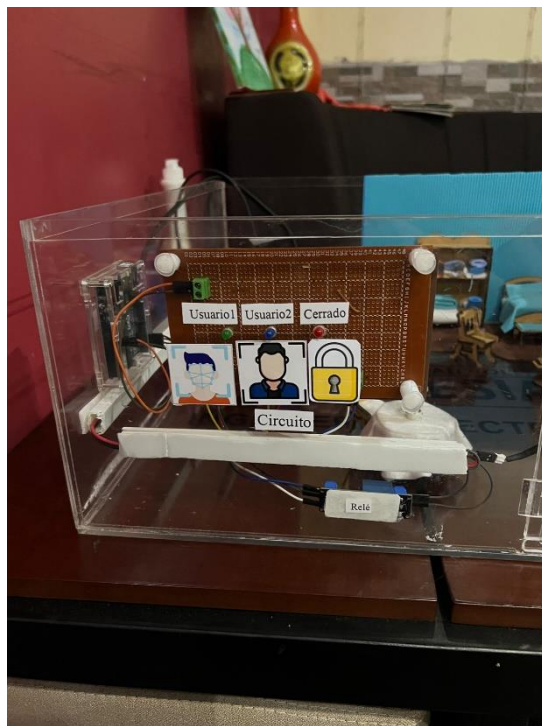
Montaje del circuito en la maqueta.



Visualización del montaje de componentes.



Etiquetado de los elementos principales.



Calcomanías para los usuarios y cerrado.



Parte frontal de la maqueta.



Parte posterior de la maqueta.