

INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA EN
INGENIERÍA Y TECNOLOGÍAS AVANZADAS

PROYECTO INTEGRADOR
INGENIERÍA EN SISTEMAS AUTOMOTRICES

ASISTENTE DE VÍDEO VIGILANCIA PARA EL AUTOMÓVIL

INTEGRANTES DEL EQUIPO:

CALVA PÉREZ SAMUEL
HERRERA RODRÍGUEZ LUIS ANTONIO
LUCIO HERNÁNDEZ CÉSAR EDUARDO

ASESORES DEL PROYECTO:

DRA. BLANCA ROSA BRISEÑO TEPEPA
ING. JULIO CÉSAR SUÁREZ TAPIA



MÉXICO, CIUDAD DE MÉXICO

OCTUBRE DE 2022

Autorización de uso de obra

Instituto Politécnico Nacional

Presente

Bajo protesta de decir verdad el que suscribe Calva Perez Samuel
(se anexa copia simple de identificación oficial), manifiesto ser autor (a) y titular de los
derechos morales y patrimoniales de la obra titulada Asistente de
video vigilancia para el automovil

en adelante "La Tesis" y de la cual se adjunta copia, por lo que por medio del presente y
con fundamento en el artículo 27 fracción II, inciso b) de la Ley Federal del Derecho de
Autor, otorgo a el Instituto Politécnico Nacional, en adelante El IPN, autorización no
exclusiva para comunicar y exhibir públicamente total o parcialmente en medios digitales,
Plataforma de la Dirección de Bibliotecas del IPN y/o consulta directa en la Coordinación
de Biblioteca de la UPIITA "La Tesis" por un periodo de 5 años contado a partir de la fecha
de la presente autorización, dicho periodo se renovará automáticamente en caso de no
dar aviso expreso a "El IPN" de su terminación.

En virtud de lo anterior, "El IPN" deberá reconocer en todo momento mi calidad de autor
de "La Tesis".

Adicionalmente, y en mi calidad de autor y titular de los derechos morales y patrimoniales
de "La Tesis", manifiesto que la misma es original y que la presente autorización no
contraviene ninguna otorgada por el suscrito respecto de "La Tesis", por lo que deslindo
de toda responsabilidad a El IPN en caso de que el contenido de "La Tesis" o la
autorización concedida afecte o viole derechos autorales, industriales, secretos
industriales, convenios o contratos de confidencialidad o en general cualquier derecho de
propiedad intelectual de terceros y asumo las consecuencias legales y económicas de
cualquier demanda o reclamación que puedan derivarse del caso.

Ciudad de México., 28 de octubre de 2022.

Atentamente

Samuel

Autorización de uso de obra

Instituto Politécnico Nacional

P r e s e n t e

Bajo protesta de decir verdad el que suscribe Lucio Hernández César Eduardo
(se anexa copia simple de identificación oficial), manifiesto ser autor (a) y titular de los
derechos morales y patrimoniales de la obra titulada Asistente de
Vídeo vigilancia para el automóvil

en adelante "La Tesis" y de la cual se adjunta copia, por lo que por medio del presente y
con fundamento en el artículo 27 fracción II, inciso b) de la Ley Federal del Derecho de
Autor, otorgo a el Instituto Politécnico Nacional, en adelante El IPN, autorización no
exclusiva para comunicar y exhibir públicamente total o parcialmente en medios digitales,
Plataforma de la Dirección de Bibliotecas del IPN y/o consulta directa en la Coordinación
de Biblioteca de la UPIITA "La Tesis" por un periodo de 5 años contado a partir de la fecha
de la presente autorización, dicho periodo se renovará automáticamente en caso de no
dar aviso expreso a "El IPN" de su terminación.

En virtud de lo anterior, "El IPN" deberá reconocer en todo momento mi calidad de autor
de "La Tesis".

Adicionalmente, y en mi calidad de autor y titular de los derechos morales y patrimoniales
de "La Tesis", manifiesto que la misma es original y que la presente autorización no
contraviene ninguna otorgada por el suscrito respecto de "La Tesis", por lo que deslindo
de toda responsabilidad a El IPN en caso de que el contenido de "La Tesis" o la
autorización concedida afecte o viole derechos autorales, industriales, secretos
industriales, convenios o contratos de confidencialidad o en general cualquier derecho de
propiedad intelectual de terceros y asumo las consecuencias legales y económicas de
cualquier demanda o reclamación que puedan derivarse del caso.

Ciudad de México., 28 de octubre de 2022.

Atentamente



Autorización de uso de obra

Instituto Politécnico Nacional

Presente

Bajo protesta de decir verdad el que suscribe Herrero Rodriguez Luis Antonio
(se anexa copia simple de identificación oficial), manifiesto ser autor (a) y titular de los
derechos morales y patrimoniales de la obra titulada Asistente de
Videovigilancia para el automovil

en adelante "La Tesis" y de la cual se adjunta copia, por lo que por medio del presente y con fundamento en el artículo 27 fracción II, inciso b) de la Ley Federal del Derecho de Autor, otorgo a el Instituto Politécnico Nacional, en adelante El IPN, autorización no exclusiva para comunicar y exhibir públicamente total o parcialmente en medios digitales, Plataforma de la Dirección de Bibliotecas del IPN y/o consulta directa en la Coordinación de Biblioteca de la UPIITA "La Tesis" por un periodo de 5 años contado a partir de la fecha de la presente autorización, dicho periodo se renovará automáticamente en caso de no dar aviso expreso a "El IPN" de su terminación.

En virtud de lo anterior, "El IPN" deberá reconocer en todo momento mi calidad de autor de "La Tesis".

Adicionalmente, y en mi calidad de autor y titular de los derechos morales y patrimoniales de "La Tesis", manifiesto que la misma es original y que la presente autorización no contraviene ninguna otorgada por el suscrito respecto de "La Tesis", por lo que deslindo de toda responsabilidad a El IPN en caso de que el contenido de "La Tesis" o la autorización concedida afecte o viole derechos autorales, industriales, secretos industriales, convenios o contratos de confidencialidad o en general cualquier derecho de propiedad intelectual de terceros y asumo las consecuencias legales y económicas de cualquier demanda o reclamación que puedan derivarse del caso.

Ciudad de México., 28 de octubre de 2022.

Atentamente





INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN
INGENIERÍA Y TECNOLOGÍAS AVANZADAS
UPIITA



ASISTENTE DE VÍDEO VIGILANCIA
PARA EL AUTOMÓVIL

QUE PARA OBTENER EL TÍTULO DE

"INGENIERO EN SISTEMAS

AUTOMOTRICES"

PRESENTAN LOS ALUMNOS:


Samuel Calva Perez
Nombre y firma del alumno


César Eduardo Lucio Hernández
Nombre y firma del alumno


Herrera Rodríguez Luis Antonio
Nombre y firma del alumno



Dra. Blanca Rosa Brizero Tepepa
Nombre y firma del asesor

ASESORES:


Inge. J. Carlos Suárez Topa
Nombre y firma del asesor

Martha Cecilia Rivas
M. en E. Martha Adoración Cecilia Rivas
Nombre y firma del sinodal

SINODALES:


Arati Rafael Canzallo Domínguez
Nombre y firma del sinodal

CIUDAD DE MÉXICO A 07 DE OCTUBRE DEL 2022



INGENIERÍA EN SISTEMAS AUTOMOTRICES

INDICE

Resumen	6
Abstract.....	7
1. Introducción.....	8
2. Planteamiento del Problema.....	10
3. Justificación.....	11
4. Objetivos.....	12
4.1. Objetivo General	12
4.2. Objetivos Específicos	12
5. Antecedentes.....	13
6. Marco Teórico	19
6.1. Raspberry Pi 4	21
6.2. Calidad de Imagen.....	22
6.3. Imagen ultra Angular	23
6.4. PCB.....	24
6.5. Sensor.....	25
6.6. Android Studio	26
6.7. Tarjeta de Desarrollo.....	27
6.8. Modulo Sim.....	28
6.9. Lenguaje KOTLIN.....	29
6.10. Reductor de voltaje	31
7. Propuesta de solución	32
7.1. Definición de la metodología	32
7.2. Desarrollo de la propuesta de solución.....	32
7.2.1. Requerimientos del asistente.....	33
7.3. Resultados esperados	33
8. Administración del proyecto	34
8.1. Presupuesto estimado e infraestructura.....	34
8.2. Planeación de Actividades	36
9. Desarrollo.....	38



INGENIERÍA EN SISTEMAS AUTOMOTRICES

9.1.	Identificación de Necesidades.....	38
9.2.	Análisis de Problema.....	39
9.3.	Definir especificaciones del diseño.....	41
9.4.	Diseño Conceptual.....	43
9.5.	Selección de una solución adecuada.....	46
9.6.	Diseño detallado del sistema	49
9.6.1.	Arduino.....	49
9.6.2.	Android Studio.....	50
9.6.3.	Raspberry.....	59
9.6.4.	PCB.....	62
9.7.	Construcción de prototipo	64
9.7.1.	Implementación del prototipo.....	72
9.8.	Costo y horas del proyecto.....	74
10.	Conclusiones	76
11.	Apéndice	77
11.1.	Código de Arduino:.....	77
11.2.	Código de Android Studio	81
11.3.	Código Raspberry pi 4.....	101
	Referencias.....	104

Imágenes

Figura 1.	Asistente de seguridad.....	20
Figura 2.	Tarjeta de desarrollo Raspberry pi 4.....	21
Figura 3.	Calidad de Imagen	23
Figura 4.	Foto Ultra Angular	24
Figura 5.	PCB.....	25
Figura 6.	Diferentes tipos de sensores	26
Figura 7.	Software de Android Studio, aplicaciones para asistente operativo Android	27
Figura 8.	Tarjetas de desarrollo	28
Figura 9.	Modulo SIM, periférico para la tarjeta Arduino	29
Figura 10.	Reductor de voltaje.....	31



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Figura 11. Modelo en V para el desarrollo de asistentes mecatrónicas.....	32
Figura 12. Mapa Conceptual, Problemas de los sistemas de seguridad	40
Figura 13. Esquema.....	43
Figura 14. Esquema 2.....	44
Figura 17. Conexión entre las Activities	51
Figura 18. Mensajes de Error Activity Registro.....	52
Figura 19. Mensaje: Cargando Perfil.....	53
Figura 20. Menú de Firebase.....	53
Figura 21. Creación de proyecto.....	54
Figura 22. Pantalla principal de tu proyecto	54
Figura 23. Menú de aplicación registrada	55
Figura 24. Menú de aplicación registrada	55
Figura 25. Habilitar proveedor.....	56
Figura 26. Usuarios Registrados.....	56
Figura 27. Datos del usuario registrado	57
Figura 28. Correo de recuperación.....	58
Figura 29. Cambio de contraseña.....	58
Figura 30. Mensaje de cerrar sesión	59
Figura 31. Mensajes de error, Activity Registro	59
Figura 32. Proyectos en Firebase... ..	58
Figura 33. Apps disponibles en firebase.....	60
Figura 34. Firebase en aplicación web	60
Figura 35. Código para agregar a código de Python	61
Figura 36. Imagen y video guardado en firebase.....	61
Figura 37. Pila y reductor de voltaje	62
Figura 38. Diseño de la PCB, pistas.....	62
Figura 39. Diseño de la PCB, Lateral.....	63
Figura 40. Diseño de la PCB, frontal	63
Figura 41. Auto sin modificaciones.....	64
Figura 42. Modificaciones al auto 1.	65
Figura 43. Modificaciones al auto 2.	65
Figura 44. Base del sensor ultrasónico y cámara.....	66



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Figura 45. Placa PCB	66
. 67	
Figura 46. Placa PCB instalada en la caja de madera	67
Figura 47. Placa PCB, Arduino, Raspberry y Modulo Sim 1.....	67
Figura 48. Placa PCB, Arduino, Raspberry y Modulo Sim 2.....	68
Figura 50. Sensores, servomotor y cámara instalados	69
Figura 51. Sensor de agua instalado.....	70
Figura 52. Reductor de voltaje y pila de moto 1.....	70
Figura 53. Reductor de voltaje y batería de moto 2	71
Figura 54. batería montada en el automóvil.....	71
Figura 55. Foto tomada por la cámara	72

Tablas

Tabla 1. Proyectos Anteriormente Elaborados.....	13
Tabla 2. Recursos humanos disponibles para el desarrollo del proyecto	34
Tabla 3. Presupuesto estimado para la realización del asistente de videovigilancia para el automóvil	35
Tabla 4. Cronograma de actividades PT1.....	36
Tabla 5. Métricas.....	42
Tabla 6. Costo de los componentes utilizados en el asistente y Horas de trabajo.....	74



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Resumen

En este proyecto se diseñará un asistente de videovigilancia para el automóvil el cual tiene como objetivo principal identificar a las personas o autos que lleguen a dañar la parte trasera de la unidad. Es decir, este asistente servirá para tener una mejor vigilancia sobre nuestro automóvil en la parte posterior.

La mayoría de las veces dejamos nuestro auto estacionado en lugares conocidos, ya sea afuera de nuestro hogar o trabajo, por lo que las personas y autos que circulan por allí concurren la zona. La propuesta que planteamos es un asistente el cual integra sensores los cuales nos permiten detectar cuando alguien se acerque al auto, además cuenta con una cámara de vigilancia con la cual podemos visualizar lo que sucede cerca del auto y finalmente un módulo SIM el cual nos alerta inmediatamente mandando un mensaje de texto al celular cuando los sensores hayan detectado actividad cerca del automóvil.

Las fotos y videos captados por la cámara se almacenarán en una memoria microSD o USB, de igual manera se almacenará en una aplicación Android y de esta forma se puede identificar al responsable que haya dañado la unidad.

En este documento se explica paso a paso la metodología que se implementó, así como las problemáticas a las que se enfrentaron, la audiencia a quien va dirigido y cómo beneficiará este asistente a la comunidad en general. Cada fase se resolverá y explicará de manera clara y ordenada con ayuda de imágenes, esquemas, tablas, etcétera.

Palabras clave: vigilancia, automóvil, sensores, identificar.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Abstract

In this project, a video surveillance assistant for the car will be designed whose main objective is to identify people or cars that damage the rear of the unit. That is, this assistant will serve to have a better surveillance of our car in the back.

Most of the time we leave our car parked in familiar places, either outside our home or work, so the people and cars that circulate their flock to the area. The proposal we propose is an assistant who integrates sensors which allow us to detect when someone approaches the car, it also has a surveillance camera with which we can visualize what is happening near the car and finally a SIM module which alerts us immediately sending a text message to the cell phone when the sensors have detected activity near the car. y.

The photos and videos captured by the camera will be stored in a microSD or USB memory, in the same way it will be stored in an Android application and in this way the person responsible for damaging the unit can be identified.

This document explains step by step the methodology that was implemented, as well as the problems they faced, the audience to whom it is addressed and how this assistant will benefit the community in general. Each phase will be resolved and explained in a clear and orderly way with the help of images, diagrams, tables, etc.

Key words: surveillance, car, sensors, identify.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

1. Introducción

¿Qué tan seguro es tener un auto en México? Es una pregunta que todos nos hacemos al comprar un nuevo auto. El robo de autos es una cuestión que se debe tener en cuenta. ¿Qué puedes hacer si tu auto no cuenta con un seguro?

En México el problema con la seguridad de autos afecta a un gran número de personas, tan solo en el 2019 se registró un robo de 84,133 unidades aseguradas, de las cuales se recuperaron 42%; además, un 60% de estos robos fueron cometidos con violencia de acuerdo con la Asociación de Compañías Mexicanas de Seguros (AMIS). [1]

Los estados que registraron una mayor incidencia de este delito fueron:

Estado de México con 22,781

Jalisco con 12,328

Ciudad de México con 9,336

Puebla con 5,533

Guanajuato con 4,711

Veracruz con 3,577

Este reporte de igual forma señala que 4 de cada 10 vehículos robados se encuentran estacionados. Y no sólo eso, muchas ocasiones igual sufrimos de rayones a la carrocería por parte de terceros los cuáles huyen y nunca sabemos quién fue el responsable.

Si eres dueño de un automóvil, contar con un seguro de auto es fundamental, ya que puede ahorrarnos muchos problemas económicos, jurídicos y médicos. Un seguro de auto puede ser la diferencia entre endeudarte por años o hasta perder tu coche y librar un imprevisto sin mayor complicación. Cuando se trate de responsabilidad y daños a la nación, las consecuencias de no tener seguro son realmente graves y trascienden de lo económico. Se debe pensar en el seguro como una inversión que lamentarás no haber hecho, si un desafortunado día sufres un siniestro. [2]

Para los propietarios de un vehículo dañado por un rayón y para las compañías de seguros de



INGENIERÍA EN SISTEMAS AUTOMOTRICES

autos, el mayor problema radica en que no exista un tercero a quién reclamar. Esto dependerá del tipo de seguro que el vehículo tenga contratado, ya que, si cuentas con una póliza de seguros contra terceros básica, los rayones de tu vehículo no se encuentran cubiertos; por ende, los gastos de reparación deberán correr por cuenta propia del dueño del vehículo. [3]

El robo de autos es sólo uno de los problemas que deberás tener en cuenta a la hora de adquirir un nuevo auto. Los rayones, abolladuras en la carrocería son más comunes al igual que el robo de autopiezas.

En el presente trabajo se plantea un asistente de vigilancia enfocado en la parte trasera de nuestro vehículo para mejorar la seguridad, teniendo en cuenta las situaciones descritas previamente. Este asistente de vigilancia nos ayudará a tener un mayor cuidado de nuestro automóvil cuando se encuentre estacionado, en caso de sufrir un rayón o golpe en la parte trasera del automóvil podremos saber quién fue el responsable y tomar las medidas necesarias.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

2. Planteamiento del Problema

El robo de autos en México es un problema que se debe solucionar, a causa de esta problemática existen varios sistemas de seguridad que, si bien no te aseguran al cien por ciento que tu auto nunca será robado, sí disminuyen considerablemente el robo.

Estos aparatos de seguridad abarcan desde un cortacorriente, alarmas y hasta la última tecnología utilizando cámaras con sensores de movimiento, asistentes RFID, GPS, etc. Evidentemente existen asistentes de seguridad más eficientes que otros, sin embargo, nuestro enfoque se basa en un asistente de vigilancia que será activado cuando nuestro auto esté estacionado.

El proyecto desarrollado plantea una solución para identificar a los responsables en caso de sufrir un rayón o daño a la unidad. Este asistente de vigilancia implementará principalmente cámaras de vigilancia, sensores y una aplicación donde podremos monitorear nuestro auto cuando estemos lejos de él. La propuesta es diseñar e implementar un prototipo que en su conjunto sea accesible y económico para que los propietarios de un automóvil puedan implementarlo.

Este asistente incluye una cámara la cual tiene un campo de visión de 120 grados. Con ayuda de los sensores ultrasónicos ésta se detiene en la dirección en que el sensor detecte un objeto acercándose. Cuando los sensores encuentren un objeto demasiado cerca durante cierto tiempo predeterminado, es decir en una zona no muy segura para el automóvil la cámara empezara a grabar el video de lo sucedido en la parte posterior del automóvil. De igual forma este asistente contará con un módulo SIM el cual nos mandará una alerta al celular, advirtiéndolo de que alguien está cerca de nuestro auto y con ayuda de una aplicación Android podremos observar las fotos o videos captados por la cámara desde nuestro dispositivo móvil.

Con este asistente de videovigilancia podremos tomar acción inmediata cuando se mande la alerta de que alguien está cerca de nuestro automóvil.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

3. Justificación

Actualmente en México los sistemas de vigilancia que existen únicamente incluyen una cámara con GPS y los precios no son muy accesibles, sin mencionar que estas cámaras tienen puntos ciegos y por ende no se tiene una visión completa del auto. El objetivo es implementar un asistente de videovigilancia el cual cubra la parte trasera del vehículo y de esta forma tener evidencia suficiente ante una autoridad competente del daño hecho al vehículo, así mismo se podría en cierta medida prevenir el robo del automóvil, al tener un monitoreo constante cuando no estemos cerca de él.

Este asistente tiene como objetivo identificar a las personas o autos que puedan dañar a la unidad cuando este estacionada. ¿Cuántas veces hemos sufrido de rayones en la carrocería? En muchas ocasiones estos daños son producidos por terceros y nunca sabemos quién fue el responsable.

La mayoría de los autos asegurados cuentan con el seguro de responsabilidad civil, el cual no se hace cargo de daños producidos por uno mismo. Sin embargo, si alguien llega a chocar tu automóvil cuando tú no estás, ¿cómo justificas ante la aseguradora que tú no fuiste el responsable?, Así pues, la cámara de vigilancia que tiene este asistente capturara las fotos o los videos de lo sucedido que a su vez servirán de evidencia.

Por lo regular siempre estacionamos nuestro auto en lugares conocidos, ya sea afuera de nuestra casa o unidad habitacional, en el estacionamiento de la escuela, trabajo, etcétera. En caso de que tu auto sufra algún daño y el responsable haya huido o simplemente no se hace responsable de lo sucedido, podrás identificar quién es el culpable y en el mejor de los casos será alguien del vecindario o conocido el cual identificarás fácilmente.



4. Objetivos

4.1. Objetivo General

Diseñar y construir un asistente de videovigilancia para un automóvil tipo sedán, para tener evidencia de siniestro ante una autoridad competente.

4.2. Objetivos Específicos

- ✚ Seleccionar los componentes necesarios para la captación de movimiento alrededor del vehículo.
- ✚ Implementar la activación de la cámara mediante los componentes seleccionados.
- ✚ Integrar físicamente los componentes electrónicos junto con la cámara.
- ✚ Seleccionar los componentes necesarios para implementar un sistema de alimentación para el asistente de video vigilancia.
- ✚ Diseñar y construir una interfaz de usuario que facilite la accesibilidad de los videos capturados por las cámaras.
- ✚ Verificar el funcionamiento del asistente de videovigilancia mediante la realización de pruebas.



5. Antecedentes

Tabla 1. *Proyectos Anteriormente Elaborados*

Número	Nombre	Descripción	Características	País	Instituto	Tipo	Referencia
1	CarCentinel	Asistente antirrobo español. Este asistente se encarga de avisar al usuario mediante una app, cuando tratan de abrir el auto, o cuando fue golpeado, etc. como también localizarlo en tiempo real, esto siendo una función muy útil, ya que varios conductores se olvidan de donde estacionaron su auto.	*Sensores de Movimientos *Modulo para enviar información a la APP. *GPS *Te permite saber más información del auto como kilómetros ha recorrido, nivel de gasolina, entre otras mas	España	CarCentinel	Producto	[4]
2	Asistente de seguridad para un Automóvil	Asistente Embebido para Vigilancia Remota de Vehículos. Se trata de un asistente que funciona para	*Raspberry pi 2 *GPS/GLONASS U-blox7 *Módulo de cámara para Raspberry Pi *Lenguaje de programación java y	España	Universidad Politécnica de Valencia	Tesis	[5]



INSTITUTO POLITÉCNICO NACIONAL
 UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



		vigilar el automóvil, ya sea cuando trata de abrirlo o lo golpean, enviando un aviso a través de una aplicación.	javascript.				
3	Asistente control de acceso a estacionamiento por medio del reconocimiento de placas vehiculares	Reconocer cuantos automóviles entran a un estacionamiento. Se encarga de observar cuantos autos entran en el estacionamiento para tener un control de los autos que entran y salen y también para evitarse robos de autos.	*Utilización del Lenguaje SQL *Visual Studio	México	IPN Zacatenco	Tesis	[6]
4	Adaptive Recognition Hungary	Reconocimiento de placas automáticamente. Su función principal de este dispositivo es controlar las placas de los autos de un lugar de	*Cámara que funciona de día y de Noche *Programa programado en Visual Studio 2010 *Además del reconocimiento de	Hungría	Adaptive Recognition Hungary	Producto	[7]



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



		trabajo, algún estacionamiento, etc, todo esto con el fin de contrarrestar los robos de autos.	placas tiene reconocimiento en rostro				
5	Asistente de seguridad automotriz 3G/GSM	Se trata de una aplicación que se encarga de alertar al conductor de cuando su automóvil fue abierto o que se lo acaban de robar, todo esto con la tecnología 3G.	*Utilización de módulo 3G *Lenguaje de programación Java *Teléfono celular con el asistente operativo de Java	México	UNAM	Tesis	[8]
6	PROCESAMIENTO DE IMÁGENES CON VISIÓN ARTIFICIAL PARA EL RECONOCIMIENTO DE PLACAS VEHICULARES	Se trata de un dispositivo que trata de cómo implementar la inteligencia artificial para el reconocimiento de imágenes, que en este caso son para el reconocimiento de placas, todo esto para el control Vehicular de algún	*Softwares Industriales IMAQ *Imaging Libraries *MATLAB	Ecuador	Universidad Técnica de Ambato	Tesis	[9]



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



		lugar.					
7	DETECCIÓN DE CARACTERES DE PLACAS DE AUTOMÓVILES MEDIANTE TÉCNICAS DE VISIÓN ARTIFICIAL	Se trata de un programa el cual se encarga de detectar los caracteres de la placa de un automóvil, ya que en ocasiones cuando se toma la foto del automóvil, no se logra percibir de una buena manera la placa, esto por la visión que no es muy buena por parte de la cámara.	*Utiliza varios softwares de imagen, esto para poder detectar la placa del automóvil *Software que tienen la herramienta de RGB.	Piura, Perú	Universidad de Piura	Tesis	[10]
8	RECONOCIMIENTO AUTOMÁTICO DE LA PLACA DE UN VEHÍCULO DE ECUADOR	Asistente que se encarga de reconocer las placas de un automóvil mediante una cámara y de esa poder tener el control vehicular Todo esto usando técnicas de visión por computador. Como	se utilizó, detección de bordes, componentes conectados y un algoritmo preliminar para la identificar la ubicación de las letras y los números de la placa. El reconocimiento automático se	Quito Ecuador	Universidad Politécnica Salesiana	Tesis	[11]



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



		parte del preprocesamiento de las imágenes se ha realizado redimensionamiento y binarización sobre las imágenes originales.	realiza mediante dos redes neuronales sobre las secciones consideradas como "caracteres".				
9	AUTOPILOT DE TESLA	capacidad de conducción autónoma total, a través de inteligencia artificial, capta automóviles, personas, etc. Atraves de cámaras, las cuales funcionan como sensores de detección.	*Cobertura con sensor avanzado: *Opera con 8 cámaras, las cuales brindan una visibilidad de 360 grados. *Las cámaras están colocadas en la parte trasera y delantera del automóvil.	USA	TESLA	Producto	[12]
10	Dash Cam	Camara que esta grabando todo el tiempo, por lo cual se puede detectar cuando alguien golpeo tu auto o tratan de abrirlo.	*Cmara ultra wide, logrando una visión de 155 grados. *La grabación no para hasta que se agote la batería o el usuario lo dese. *Suministro de	USA	Z-EDGE	Producto	[13]



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



			energía por una batería incluida, cuando se descarga comienza a utilizar la batería del automóvil. *Pantalla táctil para operar la cámara. *Visión nocturna				
11	Google Car	Es una cámara que se situa en la parte superior del automóvil, y la cual se encarga de observar lo que pasa alrededor del automóvil, esta en prueba, lo cual es un prototipo	*Solo cuenta con una cámara la cual esta girando a 360 grados, esto para que pueda ver todos los lados del automóvil. *Funciona a partir de la inteligencia artificial.	USA	GOOGLE	Prototipo	[14]



INGENIERÍA EN SISTEMAS AUTOMOTRICES

6. Marco Teórico

Un sistema de seguridad consta de 4 componentes fundamentales los cuales son, software, hardware, dispositivos periféricos y equipo de control que serán controlados por algún operador. Los diseñadores tienen la tarea de determinar el software y el hardware que se adecue a las necesidades del cliente. Esto permitirá tener un sistema que garantice que el usuario no sólo tenga confianza en el sistema, sino que además se sienta cómodo. Todo el equipo físico está controlado por programas de aplicación, estos programas pueden involucrar la implementación de una base de datos, la asignación de ciertos límites de administración a los operadores y generar un control determinado por los sensores de ambiente.

Los sistemas de seguridad han ido evolucionando conforme se van desarrollando nuevas tecnologías y los usuarios exigen mejores soluciones a sus problemas que tienen en su vida diaria, con un menor tiempo de respuesta, con mayor eficiencia y con un mínimo de errores. Hay una gran variedad de sistemas de seguridad, pueden encontrarse desde sencillos dispositivos en una red de seguridad poco compleja implementados para hogares, hasta edificios inteligentes en donde los dispositivos son capaces de tomar decisiones y se desenvuelven en un ambiente distribuido, estos son diseñados para cubrir las necesidades de empresas muy grandes.

Existen tres generaciones en la historia de los sistemas de seguridad, clasificadas dependiendo de la complejidad que involucran. La primera generación se limitaba únicamente a la implementación de un dispositivo capaz de dar aviso de cualquier violación y un medio que lo controlara; por ejemplo, se contaba tan sólo con una alarma que emitía una señal sonora cuando existía una interrupción en el esquema que tenía determinado. El control muchas veces era manual y el usuario debía de ingresar claves o parámetros para definir que la situación que se iba a presentar era del todo normal.

La segunda generación ya consistía en un medio capaz de controlar los eventos y que además podía tomar decisiones de acuerdo con la situación. Esto permitió que el usuario dejara de realizar eventos manuales y que además disminuyeran el número de falsas alarmas, pues los dispositivos eran capaces de interpretar una situación y definir si en realidad era una situación de alarma o simplemente una situación poco usual.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Por otro lado, en la tercera generación, ya se implementaron medios para poder monitorear todos los eventos que se realicen en un lugar sin que el cliente tenga que estar en la misma ubicación. Esto da flexibilidad al usuario para que al mismo tiempo que realiza otras actividades pueda estar revisando el estado en el que se encuentra la empresa o su hogar.

El objetivo de un sistema de seguridad es la detección cualquier situación de riesgo que se presente en un determinado ambiente. Estos eventos pueden variar desde la detección de un intruso hasta el reporte del inicio de un incendio. Un sistema de seguridad no significa únicamente la detección de algún problema determinado, sino también un evento como respuesta que logre poner sobre aviso a las personas correspondientes ya sea el administrador del sistema o alguna empresa dedicada a la solución de estos problemas.

Todo sistema de seguridad debe emitir una alarma que alerte a la persona responsable de cualquier desperfecto ocurrido en la empresa o en el hogar. Esta alarma, representada por cualquier tipo señal –sea sonido, imagen o texto -, debe de llegar también a una central de control ya sea dentro de la misma empresa o alguna empresa contratada externamente, para que se puedan tomar las medidas como llamar a la policía, a una agencia de seguridad o simplemente determinar que la situación no es de gravedad. [15]



Figura 1. *Asistente de seguridad [15]*



INGENIERÍA EN SISTEMAS AUTOMOTRICES

6.1. Raspberry Pi 4

Raspberry Pi es, básicamente, un ordenador. A pesar de su placa reducida, su bajo coste y su sistema operativo de código abierto Raspbian, no deja de ser un ordenador adaptado a unas necesidades de programación muy básicas. Diseñados en un principio para fomentar la enseñanza informática en las escuelas, ha tenido mucho éxito durante sus sucesivas evoluciones.

Esta Raspberry Pi 4 Modelo B es capaz de decodificar vídeo de 4K, beneficiándose de un almacenamiento más rápido a través de USB 3.0, y conexiones de red más rápidas a través de Ethernet Gigabit real.

También es la primera Raspberry Pi que admite dos pantallas a la vez, hasta dos pantallas 4K, lo que supone una gran ventaja para los creativos que desean más espacio en su escritorio.

Pero desde luego, la característica de las Raspberry que más destaca es su apariencia desnuda. Para completar el pack necesitarás de una fuente de alimentación, un monitor o un televisor, cables para conectarse al monitor (normalmente un cable micro HDMI) y un ratón y un teclado. Sin embargo, el romanticismo que implica para los amantes de la tecnología informática, el hecho de jugar con el hardware para crear tu propia máquina es algo por lo que Raspberry tiene enamorados a sus usuarios. [16]



Figura 2. Tarjeta de desarrollo Raspberry pi 4. [16]



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Características de la raspberry pi 4:

- ✚ Asistente en un chip: Broadcom BCM2711
- ✚ CPU: Procesador de cuatro núcleos a 1,5 GHz con brazo Cortex-A72
- ✚ GPU: VideoCore VI
- ✚ Memoria: 1/2/4GB LPDDR4 RAM
- ✚ Conectividad: 802.11ac Wi-Fi / Bluetooth 5.0, Gigabit Ethernet
- ✚ Vídeo y sonido: 2 x puertos micro-HDMI que admiten pantallas de 4K@60Hz a través de HDMI 2.0, puerto de pantalla MIPI DSI, puerto de cámara MIPI CSI, salida estéreo de 4 polos y puerto de vídeo compuesto.
- ✚ Puertos: 2 x USB 3.0, 2 x USB 2.0
- ✚ Alimentación: 5V/3A vía USB-C, 5V vía cabezal GPIO
- ✚ Expansión: Cabezal GPIO de 40 pines

6.2. Calidad de Imagen

No importa si hablamos de teléfonos inteligentes, de tabletas, de videocámaras deportivas tipo GoPro o de cámaras de fotografías especializadas. El primer dato -y a veces el único- que vas a ver en el resumen de sus especificaciones técnicas son los megapíxeles. Y aunque es un tema que se ha tratado en centenares de ocasiones, muchos usuarios siguen pensando que de los megapíxeles depende la calidad de las fotografías que es capaz de hacer una cámara. No es verdad. Los megapíxeles no nos dicen absolutamente nada (de calidad)

Un valor numérico, los megapíxeles, que encabezan las especificaciones técnicas de la mayoría de los dispositivos capaces de hacer fotos. Este valor nos indica el tamaño del sensor, la cantidad de puntos del sensor y, por tanto, la resolución máxima. Ahora bien, lo que define la calidad de una fotografía es el color, la luz, el contraste, la compresión y otros muchos valores que poco o absolutamente nada tienen que ver con los megapíxeles. En pocas palabras, una buena foto depende de la luz

Si consultas con cualquier fotógrafo profesional, la respuesta no da lugar a dudas: la calidad depende de la luz. Quiere esto decir que, aunque tengamos la mejor cámara del mercado, con el mejor sensor y las mejores lentes en su objetivo, así como el procesador más potente y "todo eso", sin un buen tratado o uso de la luz no vamos a conseguir más que lo que es capaz de sacar



INGENIERÍA EN SISTEMAS AUTOMOTRICES

un buen fotógrafo de la sencilla cámara de un móvil. [17]

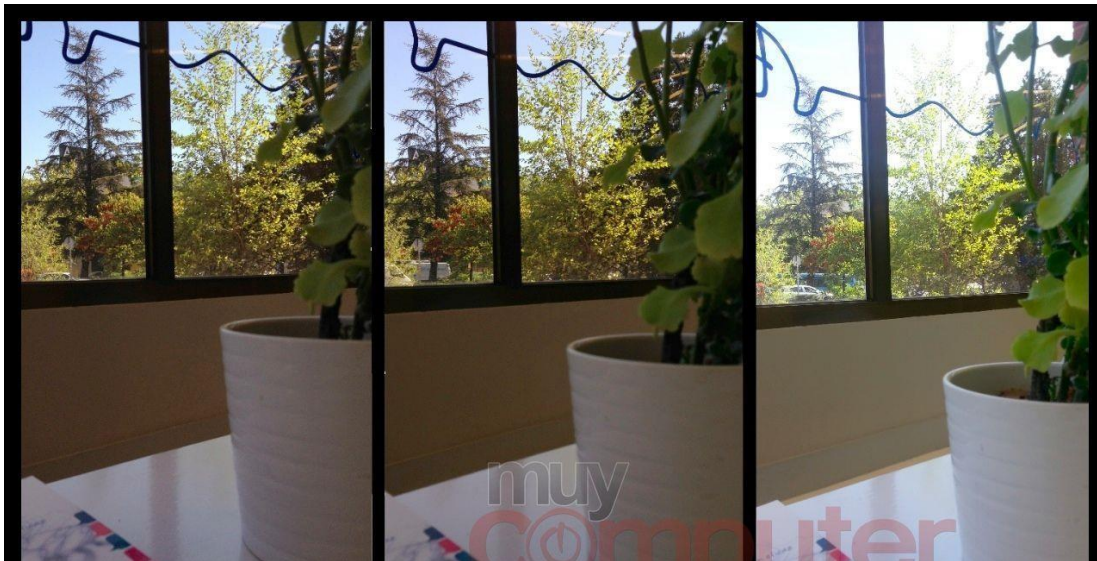


Figura 3. *Calidad de Imagen* [17]

6.3. Imagen ultra Angular

Cuando se habla de amplitud de la foto tomada, se refiere a cuantos objetos puede captar la cámara fotográfica, esto quiere decir que sea una cámara ultrawide, la cual se refiere a que tomara una foto como si se comportara como una panorámica.

El principal objetivo de una lente gran angular es la de lograr una mayor profundidad de campo, es decir, cuando el usuario quiere que en la imagen aparezca reflejada la mayor parte posible del objeto a fotografiar, cuando este es distante o amplio. Al provocar una distorsión cuando el objetivo está cerca, también suele ser utilizado para lograr efectos artísticos. No obstante, el objetivo de la mayoría de los usuarios es el de poder captar paisajes más ampliamente que con el objetivo estándar.

Los ángulos de visión de una lente gran angular pueden variar entre los 60° y los 180°, mientras que, sin haber una regla fija, la distancia focal es inferior a los 35mm. Si fuese mejor a los 24mm se considera una lente ultra gran angular, que también comienza a integrar en muchos de los móviles que hemos visto en 2019. Es decir, que algunos móviles tienen una lente gran angular, y una lente ultra gran angular y permiten variar entre ellas dentro de las opciones de la cámara, previas al disparo. Muchos fabricantes, explican los resultados del gran angular, al campo de visión del ojo humano, para hacer entender las perspectivas y los resultados que se obtienen



INGENIERÍA EN SISTEMAS AUTOMOTRICES

con estas lentes. [18]



Figura 4. Foto Ultra Angular [18]

6.4. PCB

Una tarjeta de circuito impreso es un circuito cuyos componentes y conductores están contenidos dentro de una estructura mecánica. Las funciones conductoras incluyen trazas de cobre, terminales, disipadores de calor o conductores planos. La estructura mecánica se hace con material laminado aislante entre capas de material conductor. La estructura general es chapada y está cubierta con una máscara de soldadura no conductora y una pantalla de impresión para la ubicación de leyenda de componentes electrónicos.

La tarjeta de circuito impreso está construida por capas que se alternan de cobre conductor con capas de material aislante no conductor. Durante la fabricación, se graban las capas de cobre internas dejando trazas de cobre intencionadas para conectar los componentes de circuito. Una vez laminado el material de aislamiento es grabado a las capas de cobre y así sucesivamente hasta que la tarjeta de circuito impreso esté completa

Los componentes se agregan a las capas externas de la tarjeta de circuito impreso cuando todas las capas se han grabado y laminado juntas. Las partes de montaje superficial se aplican



INGENIERÍA EN SISTEMAS AUTOMOTRICES

automáticamente con robots y las partes con orificio pasante se colocan manualmente. Luego, todas las partes se sueldan en la tarjeta utilizando técnicas tales como reflujo o soldadura por ola. El montaje final se chapa después de que se aplican la máscara de soldadura y la pantalla de impresión de la leyenda. [19]

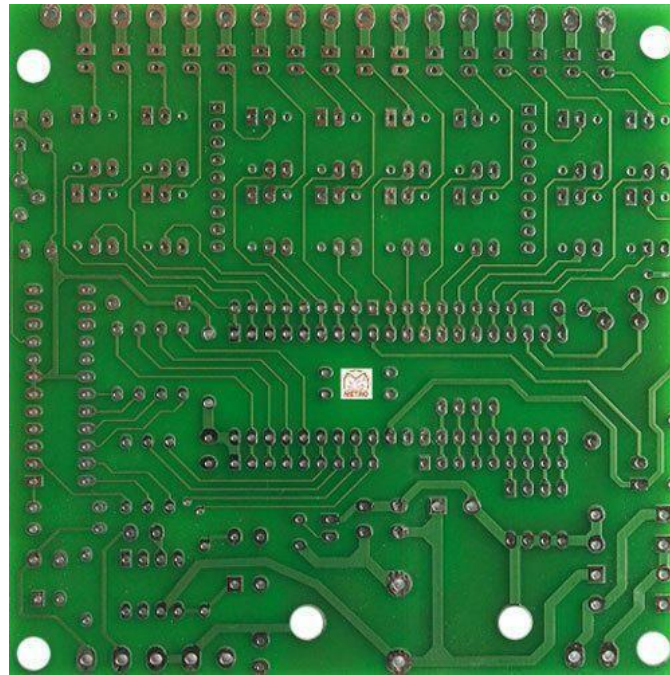


Figura 5. PCB [19]

6.5. Sensor

Un sensor es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas.

- Las variables de instrumentación pueden ser, por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, etc.
- Una magnitud eléctrica puede ser una resistencia eléctrica (como en una RTD), una capacidad eléctrica (como en un sensor de humedad o un sensor capacitivo), una tensión eléctrica (como en un termopar), una corriente eléctrica (como en un fototransistor), etc.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Los sensores se pueden clasificar en función de los datos de salida en:

✚ Digitales ✚

Analógicos

Dependiendo del tipo de sensor, debemos conectarlo a una entrada digital o analógica.

Y dentro de los sensores digitales, estos nos pueden dar una señal digital simple con dos estados como una salida de contacto libre de tensión o una salida en bus digital. [20]

Figura 6. *Diferentes tipos de sensores [20]*



6.6. Android Studio

Android Studio es un entorno de desarrollo, un software, que cuenta con herramientas y servicios para que los desarrolladores puedan crear nuevas aplicaciones para Android.

En este entorno de desarrollo nos encontraremos con las herramientas que son necesarias para crear aplicaciones. Esto incluye desde el código, al diseño de la interfaz de usuario de la aplicación. [21]



INGENIERÍA EN SISTEMAS AUTOMOTRICES

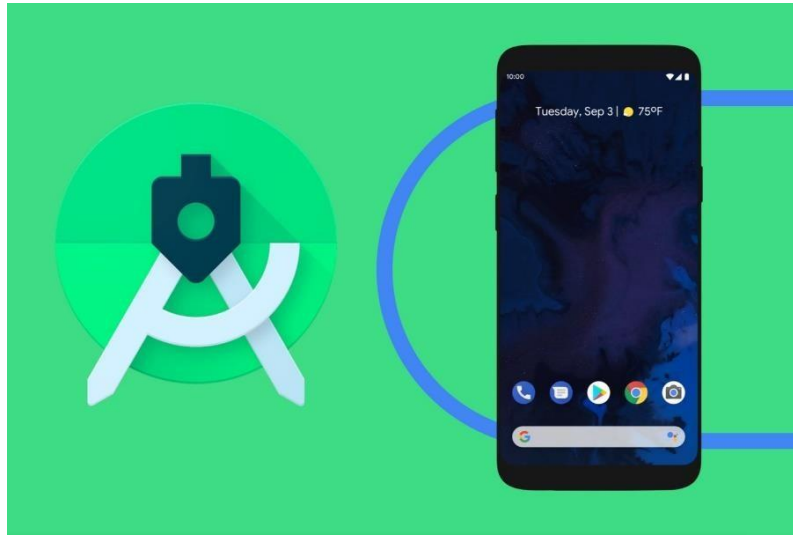


Figura 7. *Software de Android Studio, aplicaciones para asistente operativo Android [21]*

6.7. Tarjeta de Desarrollo

La historia de las tarjetas de desarrollo microcontrolador comercial comienza en los 70S, con la llegada del Intel 4004 de 4 bits. Fue el segundo CPU completo de un solo chip de la historia, y el primero en estar disponible comercialmente.

Pero fue el microcontrolador PIC de Microchip Technology, que data de 1975, el que se convirtió en la columna vertebral del mercado de los makers durante muchos años, debido a una combinación de factores que incluyen bajo costo, fácil disponibilidad, y la proliferación de herramientas de programación libres.

Una tarjeta de desarrollo es una placa o circuito que contiene un microcontrolador principal que corre o ejecuta una serie de instrucciones de un programa suministrado. Alrededor de este procesador o unidad principal se ha creado un diseño electrónico que permite: la programación del componente suministra el voltaje adecuado para el correcto funcionamiento del controlador y proporciona acceso a las entrada y salidas del microcontrolador para la conexión de sensores y actuadores.

Las placas de desarrollo incluyen cosas como una fuente de alimentación, soporte para conectar sensores y, a veces, incluso sensores y actuadores a bordo. También permiten a los usuarios



INGENIERÍA EN SISTEMAS AUTOMOTRICES

conectar rápidamente sensores y actuadores (si no están ya incluidos en la placa) y el software que los acompaña facilita la creación y el despliegue del código.

Hay muchos tableros de desarrollo y microcontroladores disponibles de una variedad de compañías: TI, Samsung, Arduino, Raspberry Pi y más. Elegir cuál es el adecuado para usted depende de una serie de factores que varían según la naturaleza de su aplicación. [22]

Como se menciona antes, actualmente hay una gran variedad de tarjetas de desarrollo, cada una ofrece características especiales. Hay tarjetas enfocadas a:

- ✚ Diseño electrónico.
- ✚ Dispositivos de IoT (Internet de las cosas).
- ✚ Dispositivos portables (wearables).
- ✚ Dispositivos con Inteligencia Artificial.

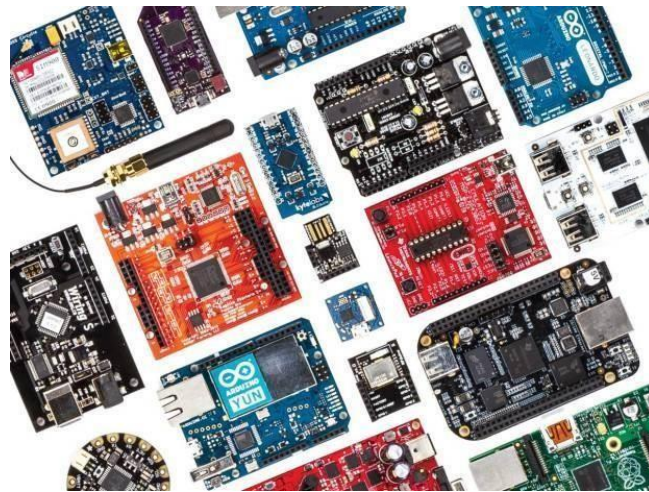


Figura 8. Tarjetas de desarrollo [22]

6.8. Modulo Sim

EL GSM (Global System for Global Communications) es el asistente de comunicaciones que más se utiliza en teléfonos móviles y es un estándar en Europa. La primera funcionalidad de esta tecnología es la transmisión de voz, pero también permite la transmisión de datos (SMS, Internet), eso sí, a una velocidad muy baja de 9kb/s.

El GPRS (General Packet Radio Service) es una extensión del GSM basada en la transmisión por paquetes que ofrece un servicio más eficiente para las comunicaciones de datos,



INGENIERÍA EN SISTEMAS AUTOMOTRICES

especialmente en el caso del acceso a Internet. La velocidad máxima (en teoría) del GPRS es de 171kb/s, aunque en la práctica es bastante más pequeña. [23]



Figura 9. *Modulo SIM, periférico para la tarjeta Arduino [23]*

6.9. Lenguaje KOTLIN

Kotlin es un lenguaje de programación para el desarrollo de aplicaciones Android y Android Studio. Se caracteriza por ser expresivo y conciso, y está diseñado para proveer seguridad de tipos y contra nulos.

El mismo fue creado en 2011 por los programadores de JetBrains, quienes en busca de sustituir el lenguaje de programación Java en sus proyectos Android, idearon un lenguaje muy similar al de Scala, capaz de lograr una excelente funcionalidad durante la programación con la claridad y seguridad que se necesita para hacerlo.

Tal fue su éxito, que, en 2012, el lenguaje Kotlin pasó a convertirse en un proyecto de código abierto bajo la licencia Apache, a la cual pertenece también Android y Android Studio. Para el 2017, Kotlin pasaría a ser reconocido oficialmente, ni más ni menos, que por Google para el desarrollo de aplicaciones Android, al mismo nivel que el lenguaje Java.

el lenguaje de programación Kotlin se caracteriza por su exactitud y claridad, lo que permite reducir notablemente los errores comunes de código y llevar a cabo, sin inconveniente alguno, las tareas de desarrollo en cada una de las aplicaciones.

Kotlin vs. Java



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Tal como te hemos contado, el lenguaje de programación Kotlin fue diseñado con el objetivo de reemplazar al lenguaje Java, dado que su uso resulta mucho más sencillo, ágil y efectivo. Esto, sin duda, tuvo mucha más repercusión cuando Google declaró a Kotlin como el lenguaje oficial de Android, igualándolo a Java y C++.

Por este motivo, Android Developers ha permitido, desde un principio, la compatibilidad entre estos dos lenguajes para que, quienes todavía no saben qué es Kotlin y cómo funciona, puedan probarlo y usarlo fácilmente en las aplicaciones de sus dispositivos.

Dicho esto, si lo que quieres es integrar de alguna manera Kotlin con Java, podrás llevar a cabo varias acciones, tales como:

- ✚ Hacer una migración Java a Kotlin.
- ✚ Usar Kotlin en un proyecto ya empezado con código Java.
- ✚ Realizar ciertas tareas de programación en Kotlin y otras en Java, es decir, usar ambos lenguajes de programación a la vez.
- ✚ Incorporar archivos Kotlin y Java en un mismo proyecto.
- ✚ Llamar al código Java desde Kotlin, o viceversa. Así, podrás aprovechar todas tus bibliotecas Java ya existentes.

Exactitud y claridad

Una de las principales características de este lenguaje de programación para Android, es su nivel de exactitud y claridad, ya que reduce notablemente la repetición del código, permitiendo así, que los desarrolladores y programadores ahorren tiempo y recursos.

Del mismo modo, también disminuye la posibilidad de cometer algún error, y en tal caso, si ocurriera, el mismo sería más sencillo de detectar. Según Android Developers, al momento de explicar qué es Kotlin y sus ventajas, afirma que todas aquellas aplicaciones para Android que contienen el código Kotlin tienen un 20% menos de posibilidades de fallar.

Simultaneidad asegurada

Gracias a las cortinas de Kotlin, podrás trabajar, de manera sencilla, tanto con código de bloqueo como con código asíncrono, ya que las mismas se encargan de simplificar en segundo plano toda la administración de tareas.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Desventajas:

- ✚ La coincidencia de sus patrones es algo débil.
- ✚ No cuenta con las últimas versiones de plugins de Gradle (especialmente Alfa o Beta).
- ✚ Al principio, la legibilidad inicial del código resulta un poco difícil de entender.
- ✚ Su tiempo de ejecución es, a veces, un poco más lento en comparación con el lenguaje Java.
- ✚ Al ser un lenguaje de programación relativamente nuevo, su comunidad de soporte es muy pequeña en comparación a otras. [24]

6.10. Reductor de voltaje

Un regulador de voltaje es un dispositivo eléctrico diseñado para suministrar un voltaje estable y proteger a los equipos eléctricos y electrónicos conectados a una línea de tensión contra fenómenos como sobrevoltaje, caída de tensión y variaciones de voltaje.

Ya que todos los equipos y maquinaria eléctrica están diseñados para una operación correcta dentro de ciertos parámetros de voltaje predeterminados de fábrica, si los valores máximos o mínimos de alimentación de voltaje eléctrico son excedidos existe un riesgo latente de fallas debido a las condiciones de operación anormales.

Un regulador de voltaje es capaz de recibir un voltaje de entrada variable y busca proveer un voltaje de salida apropiado para el funcionamiento correcto de los equipos conectados como carga a la salida de este.

Existen diferentes tecnologías y tipos de reguladores de voltaje con diferentes especificaciones técnicas y con ventajas y desventajas comparativas, sin embargo, el objetivo principal de cualquier regulador es mantener el voltaje estable en una línea eléctrica para evitar daños y fallas en los equipos energizados. [25]



Figura 10. Reductor de voltaje [25]

INGENIERÍA EN SISTEMAS AUTOMOTRICES

7. Propuesta de solución

7.1. Definición de la metodología

Para la elaboración de este proyecto nos basaremos en una metodología especializada, utilizaremos el modelo en V creado por Paul Rook, el cual se enfoca en mejorar la eficiencia del software a desarrollar implementando un proceso de verificación en cada fase para el desarrollo del producto, en vez de ponerlo a prueba al final. Este proceso parte desde las necesidades del cliente hasta la implementación del asistente funcionando.

A continuación, se muestra un esquema del proceso que se llevará a cabo a lo largo de este proyecto. Como se puede observar se tendrán que seguir varias fases antes de conseguir el resultado deseado.

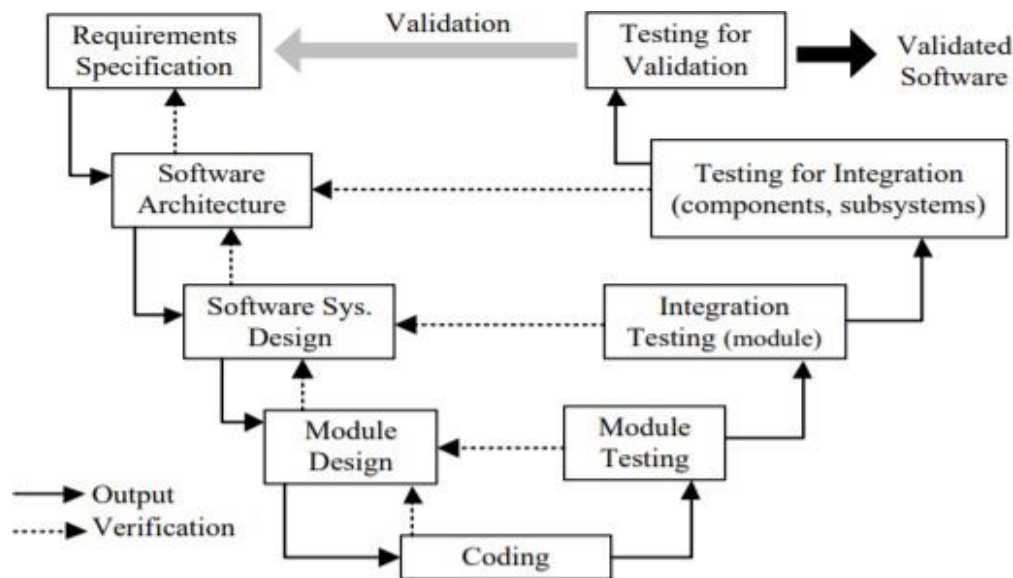


Figura 11. Modelo en V para el desarrollo de asistentes mecánicas.

7.2. Desarrollo de la propuesta de solución

- ✚ El asistente propuesto en este proyecto deberá llevar a cabo las siguientes funciones.
- ✚ Identificación de autos aproximándose a nuestro vehículo.
- ✚ Captar en vídeo a las personas o autos que se hayan acercado al vehículo.
- ✚ Notificar al usuario por su celular que se ha detectado a alguien cerca de su auto.
- ✚ Subir la evidencia registrada a un espacio de almacenamiento.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

7.2.1. Requerimientos del asistente

El asistente contara con sensores y actuadores los cuales ayudaran a la detección de objetos, tratando de monitorear la parte trasera del automóvil.

De igual forma se implementará un módulo de emisión - recepción, el cuál puede ser del tipo WI-FI o SIM, éste servirá para notificar al usuario que se ha detectado actividad cerca de su vehículo.

Finalmente se desarrollará una aplicación con interfaz atractiva e intuitiva en la cual se podrá visualizar el vídeo. Esta aplicación tendrá diferentes funciones, de las cuales la que más destaca será el acceso y almacenamiento del contenido multimedia que se generará cuando el automóvil este estacionado.

7.3. Resultados esperados

Al haber concluido este proyecto se espera que sea accesible y eficaz para los autos sedan. Este asistente de vigilancia podrá ser capaz de detectar actividad que ocurra cerca de la parte posterior del vehículo, alertando al dueño desde la comodidad de su celular y de esta forma, identificar a la persona o automóvil que llegue a dañar a la unidad, como pueden ser rayones o daños severos. Además, posiblemente contaremos con evidencia del robo de autopartes de nuestro automóvil.

Como se mencionó previamente, la mayoría de las veces dejamos nuestro auto estacionado en lugares conocidos, ya sea afuera de nuestra casa o en el estacionamiento del trabajo, por lo que las personas y autos que circulan por allí son, en el mayor de los casos, del vecindario y son fáciles de reconocer. El asistente detectará a estas personas en caso de dañar la unidad, y con las evidencias se podrá identificar al responsable. La aplicación móvil será de gran ayuda, ya que con ésta podremos ser notificados cuando alguien esté cerca de nuestro auto y poder tomar acción de inmediato.

Finalmente, en caso de que el auto haya sufrido algún daño, el vídeo captado servirá como evidencia para nuestra aseguradora, ya que algunas no se hacen cargo por el daño causado por nosotros mismos, con esto podremos demostrar nuestra inocencia, deslindar responsabilidades y que la misma aseguradora define el grado de culpabilidad de los responsables.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

8. Administración del proyecto

8.1. Presupuesto estimado e infraestructura

En la siguiente tabla se muestran los recursos humanos disponibles para la realización del proyecto **Tabla 1**, por otra parte, los recursos previstos para la gestación del proyecto y la culminación de este se pueden ver enlistados en la **Tabla 2** que desglosa el presupuesto estimado el cual asciende a \$5,700.00 MXN. Es importante aclarar que los procesos propuestos y los materiales están sujetos a cambios debido a los diferentes materiales, técnicas, dimensiones y funcionalidades aun no definidas.

Tabla 2. Recursos humanos disponibles para el desarrollo del proyecto

Recursos humanos	STEM	Institución	Tiempo destinado
Estudiante Calva Pérez Samuel	STEM	IPN-UPIITA	680 horas
Estudiante Herrera Rodríguez Luis Antonio	STEM	IPN-UPIITA	680 horas
Estudiante Lucio Hernández Cesar Eduardo	STEM	IPN-UPIITA	680 horas
Asesor Dra. Briseño Tepepa Blanca Rosa	STEM	IPN-UPIITA	68 horas
Asesor Ing. Julio Cesar Suarez Tapia	STEM	IPN-UPIITA	68 horas
Tiempo total			2108 horas



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Tabla 3. *Presupuesto estimado para la realización del asistente de videovigilancia para el automóvil*

Modulo/Recurso	Recurso material	Cantidad	Costo	Fuente de financiamiento
Asistente de visualización	Cámaras	2	\$2,000.00	Propio
Asistente de captación de objetos	Sensores Actuadores	1	\$2,000.00	Propio
Estructura o carcasa	Lamina para carcasa, sujetadores.	1	\$300.00	Propio
Asistente de alimentación	Material eléctrico y electrónico	1	\$1,000.00	Propio
Asistente de distribución energética	Material eléctrico y electrónico.	1	\$400.00	Propio
Costo total			\$5,700.00	



INSTITUTO POLITÉCNICO NACIONAL
 UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



8.2. Planeación de Actividades

Las actividades que conducirán el desarrollo del proyecto, son mostradas en la Tabla 4.

Tabla 4. Cronograma de actividades PT1

Id	Nombre de Tarea	Duración	Responsable	Semana Marzo-Junio																
				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	2. Implementación	17 Semanas																		
2	2.1. Programar y configurar los sensores	4 semanas																		
3	2.1.1. Configurar sensor de movimiento	1 semana	Cesar Eduardo Lucio Hernández																	
4	2.1.2. Programación de los sensores de movimiento en Arduino	1 semana	Cesar Eduardo Lucio Hernández																	
5	2.1.3. Programación del módulo SIM en Arduino	2 semanas	Cesar Eduardo Lucio Hernández																	
6	2.2. Desarrollo de la aplicación	6 semanas																		
7	2.2.1. Programación de la aplicación en Android Studio	6 semanas	Luis Antonio Herrera Rodriguez																	
8	2.3. Configuración de la cámara	7 semanas																		
9	2.3.1. Pruebas de las fotos obtenidas con la cámara	1 semanas	Samuel Calva Pérez																	
10	2.3.2. Procesamiento de imagen en el software	3 semanas	Samuel Calva Pérez																	
11	2.3.3. Programación para la calidad de las imágenes	3 semanas	Samuel Calva Pérez																	
12	2.4. Vinculación entre el Hardware y el Software	4 semanas																		
13	2.4.1 Vinculación de la cámara con Arduino	1 semana	Samuel Calva Pérez																	



INGENIERÍA EN SISTEMAS AUTOMOTRICES

9. Desarrollo

Para el desarrollo de este proyecto Integrador, se debió seguir una metodología de diseño, la cual se basa de la siguiente manera:

- + Identificación de Necesidades
- + Análisis del Problema
- + Definir especificaciones de diseño deseable
- + Diseño Conceptual
- + Selección de una solución adecuada
- + Diseño detallado del asistente (Arduino, Raspberry: Python, Android Studio: Kotlin)
- + Documentación
- + Implementación del Prototipo
- + Pruebas

Estos pasos son fundamentales para cualquier tipo de proyecto.

9.1. Identificación de Necesidades

En la sociedad mexicana siempre se ha detectado el mismo problema durante años, el cual es la inseguridad que tienen los automóviles. En ocasiones nos ha tocado sufrir un golpe en nuestro automóvil, provocando que tengamos que gastar dinero, sin saber quién fue el culpable. Esto sucede con mayor frecuencia en plazas comerciales, como también en unidades habitacionales de la ciudad de México.

Es por esa razón que es una muy buena idea implementar un asistente que nos tenga al tanto de la seguridad de nuestro automóvil, de esa manera el conductor o dueño del automóvil, tenga esa tranquilidad de saber que su auto está seguro.

De igual manera como sabemos hoy en día, gran parte de la población tiene un teléfono inteligente, el cual lo lleva a todas partes, así que sería buena idea, que este asistente que pensamos implementar se pudiera manejar desde el teléfono, la seguridad de tu auto, al alcance de tu mano.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

9.2. Análisis de Problema

Como ya se mencionó anteriormente, en muchas ciudades grandes del país, como lo es la ciudad de México, cuando un conductor va a una plaza cercana, en ocasiones este llega a tener miedo de que golpeen su auto, esto porque en ocasiones los conductores no cuentan con la pericia para estacionarse bien, ocasionando que golpeen a otro automóvil o por otras circunstancias suceden este tipo de accidentes, esto mismo sucede en las unidades habitacionales, en ocasiones como los automóviles quedan muy cerca del uno al otro, los conductores llegan a golpearlos, provocando que la persona que golpeo al automóvil huya, y lo que provoca todo esto, es que el dueño del auto tenga que gastar dinero, el cual no estaba en sus planes.

Hemos visto de igual manera que hay dispositivos de seguridad para automóvil, sin embargo, estos llegan a tener un costo muy elevado, además de sus fallas que llegan a tener en ocasiones o las limitantes que tienen. Además, hemos visto que muchas veces las aplicaciones que tienen estos asistentes no llegan a ser muy fáciles de manejar para el usuario, como también los errores que llegan a tener a la hora de estar funcionando.

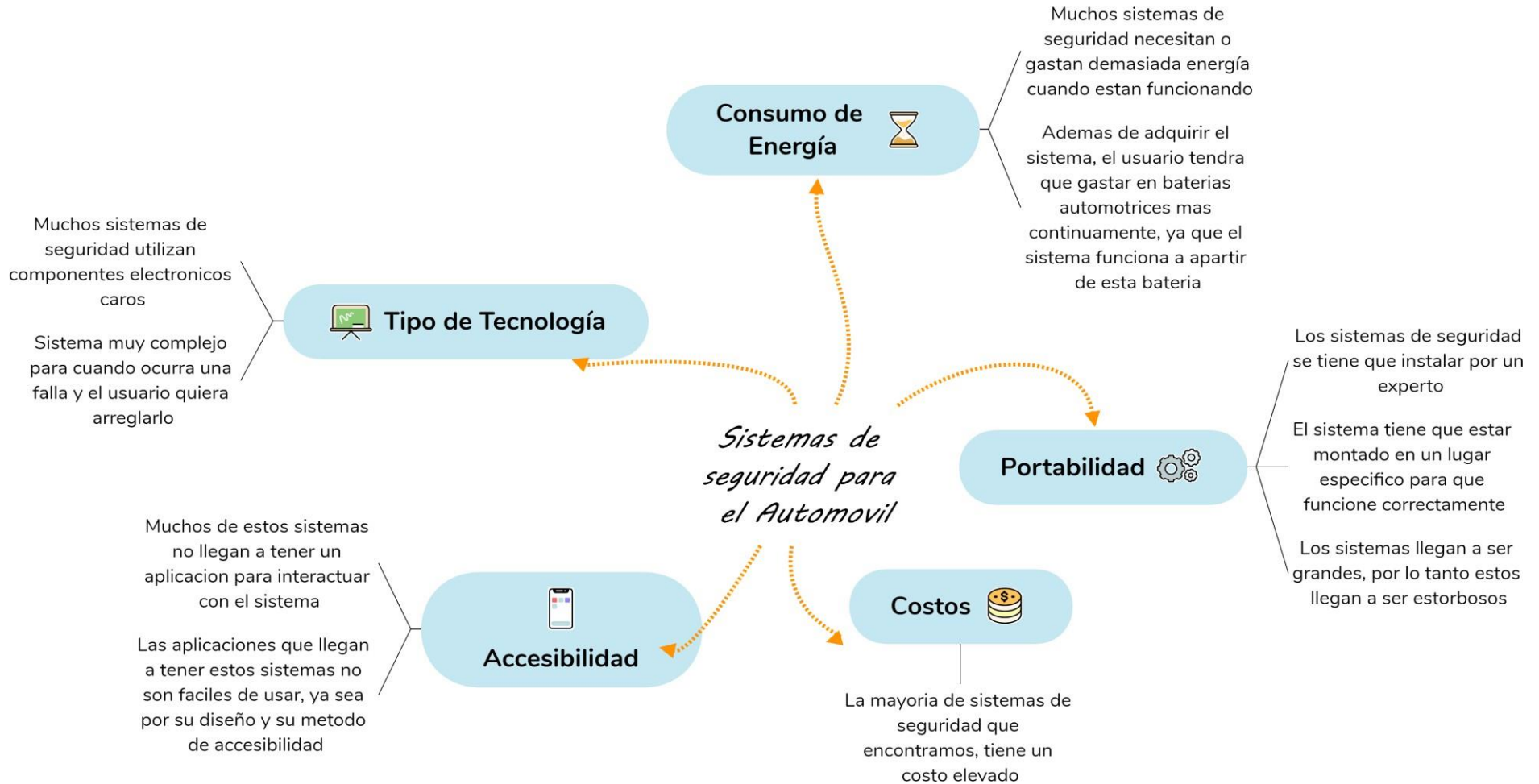
Además, hoy en día los automóviles montan tanta tecnología de seguridad que a veces se vuelve complicado entender que son y cómo pueden ayudarnos. De inicio, debemos saber que los sistemas de seguridad de un automóvil se dividen en dos categorías, unos son los sistemas activos, es decir, aquellos que están en funcionamiento en todo momento y cuyo objetivo es el de ayudarnos a evitar el accidente. El segundo tipo de sistemas son los de seguridad pasiva que, a diferencia de los activos, su objetivo es salvar la vida y proteger a los ocupantes de lesiones una vez que el accidente ya ha ocurrido o es inevitable que suceda. De tal forma se podría decir que nuestro proyecto sería un sistema activo, ya que nuestro asistente de videovigilancia estaría en funcionamiento en todo momento para capturar el momento de algún incidente en el automóvil.

Muchos de los sistemas de seguridad de este tipo, que sería antigolpes, llegan a tener fallas a la hora de su funcionamiento, como por ejemplo con las cámaras, ya que estas en ocasiones no llegan a ser de buena calidad, perdiendo claridad a la hora de tomar una foto o un video, provocando que no se logre ver con claridad el auto que golpeo el nuestro. Como también el rango de amplitud, las cámaras llegan a tener una amplitud muy corta, lo que conlleva que en ocasiones se tome la foto, pero esta no logre captar todo el automóvil.



A continuación, se muestra los problemas de los sistemas de seguridad:

Figura 12. Mapa Conceptual, Problemas de los sistemas de seguridad





9.3. Definir especificaciones del diseño

Cada persona que posee un vehículo se enfrenta al mismo desafío, el cual es protegerlo y cuidar de él. Desde hace más de una década se ha estado diseñando e implementando sistemas de protección y seguridad para el vehículo, y de esta forma poder disminuir el robo de unidades. Actualmente en el siglo XXI la tecnología empleada para la seguridad de vehículos se ha enfocado en los sistemas automatizados los cuales no se necesita de una persona para poder activarlos. Estos tienen su fundamento en la electrónica y programación, de ahí surgen los microcontroladores los cuales son la base de cualquier sistema de seguridad moderno.

Los microcontroladores también conocidos como placas de desarrollo pueden conseguirse fácilmente. Existen muchos tipos de placas de desarrollo. El Arduino uno tiene el microcontrolador ATMEGA328 el cual es donde se programa y almacenan las sentencias de código deseadas y de esta forma podemos construir nuestros proyectos de una forma intuitiva y sencilla. Arduino posee un entorno de desarrollo en el cual facilita la programación del atmega328, llamado Arduino IDE.

Existen muchos tutoriales y cursos en internet que nos explican a detalle cómo programar en esta plataforma con Arduino, de igual forma existe bastante información en libros que nos ayudan a comprender el funcionamiento de Arduino, el cual está basado en C++.

Los sistemas de seguridad más modernos están diseñados con placas de desarrollo, como por ejemplo los microcontroladores de la familia de ATMEL y hay algunos que utilizan otras placas de desarrollo, como pueden ser STM32 o Raspberry pi. Estos sistemas de seguridad incluyen sensores y actuadores, los sensores se encargan de detectar magnitudes físicas como pueden ser temperatura, presencia, peso, etc. Pero los sensores más utilizados para la seguridad son los sensores de presencia y movimiento. Estos sensores nos ayudan a detectar personas o vehículos que lleguen a estar cerca de nuestro vehículo y en ese momento esa información se procesa y se puede activar una alarma, una cámara, un mensaje de advertencia, etc.

En este proyecto se utilizará la tarjeta de desarrollo de Arduino nano debido a que cumple con los requerimientos necesarios para la implementación del asistente, y además por ser la tarjeta de desarrollo más accesible, económica y fácil de programar no sólo en México sino en todo el mundo.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

A continuación, se listan las necesidades generales requeridas para este proyecto:

- ✚ Prototipo eficiente.
- ✚ Intuitivo para el usuario.
- ✚ Económico.
- ✚ Bajo consumo de energía.
- ✚ Moderno y atractivo.
- ✚ Espacio de trabajo pequeño

Características (métricas) deseables:

- ✚ Configurable (el usuario podrá ajustarlo a sus necesidades).
- ✚ Materiales seguros y confiables
- ✚ Intuitivo (fácil de usar).
- ✚ Efectivo (cumplir los objetivos con los recursos y materiales obtenidos).
- ✚ Económico (asistente de alta confiabilidad y más económico que los del mercado).
- ✚ Dimensiones del asistente (las placas de desarrollo ocuparán un área de trabajo de 9 x 15cm)

En la siguiente tabla (tabla 5) se hace referencia a las necesidades mencionadas anteriormente para poder visualizar y entender de una mejor forma la importancia de cada una de ellas, donde el 3 representa la mayor importancia.

Métrica No.	Métrica	Necesidades	Importancia	Unidades	Valor marginal	Valor ideal
1	Configurable	2	2	%	>85	90
2	Materiales seguros y confiables	1, 6	2	%	>85	90
3	Intuitivo	2, 1	3	%	>95	95
4	Efectivo	1, 4	3	%	>90	95
5	Económico	3	2	MXN	<3000	2500
6	Dimensiones del asistente	6	2	Cm ²	<400	250

Tabla 5. Métricas



INGENIERÍA EN SISTEMAS AUTOMOTRICES

9.4. Diseño Conceptual

Una vez definidas las necesidades del asistente podemos enfocarnos en los conceptos de diseño y las soluciones a implementar, de esta forma podremos entender mejor los métodos que se llevarán a cabo.

Recordando que el objetivo de este proyecto es diseñar un asistente de videovigilancia para el automóvil el cual podrá detectar a personas o vehículos que se acerquen al nuestro y en ese momento alertar al propietario que se ha detectado actividad sospechosa cerca de la unidad. Si alguna persona o automóvil llega a dañar, las cámaras serán las encargadas de grabar lo ocurrido gracias a los sensores, y con ayuda del módulo SIM el cual será el encargado de notificar al propietario, éste podrá tomar las acciones necesarias.

Ahora podemos analizar los siguientes esquemas.

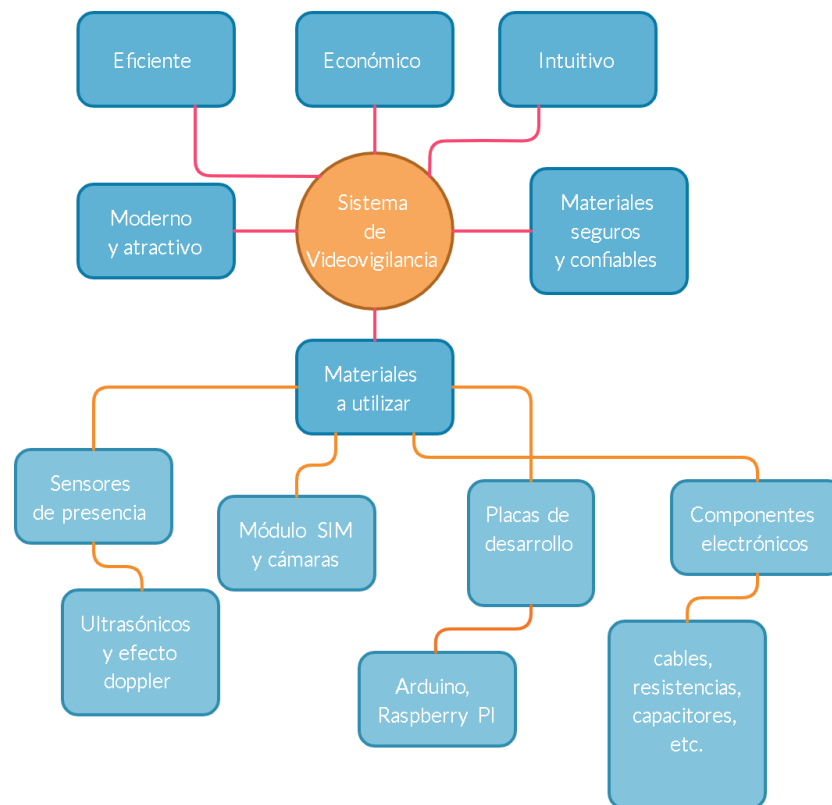


Figura 13. Esquema

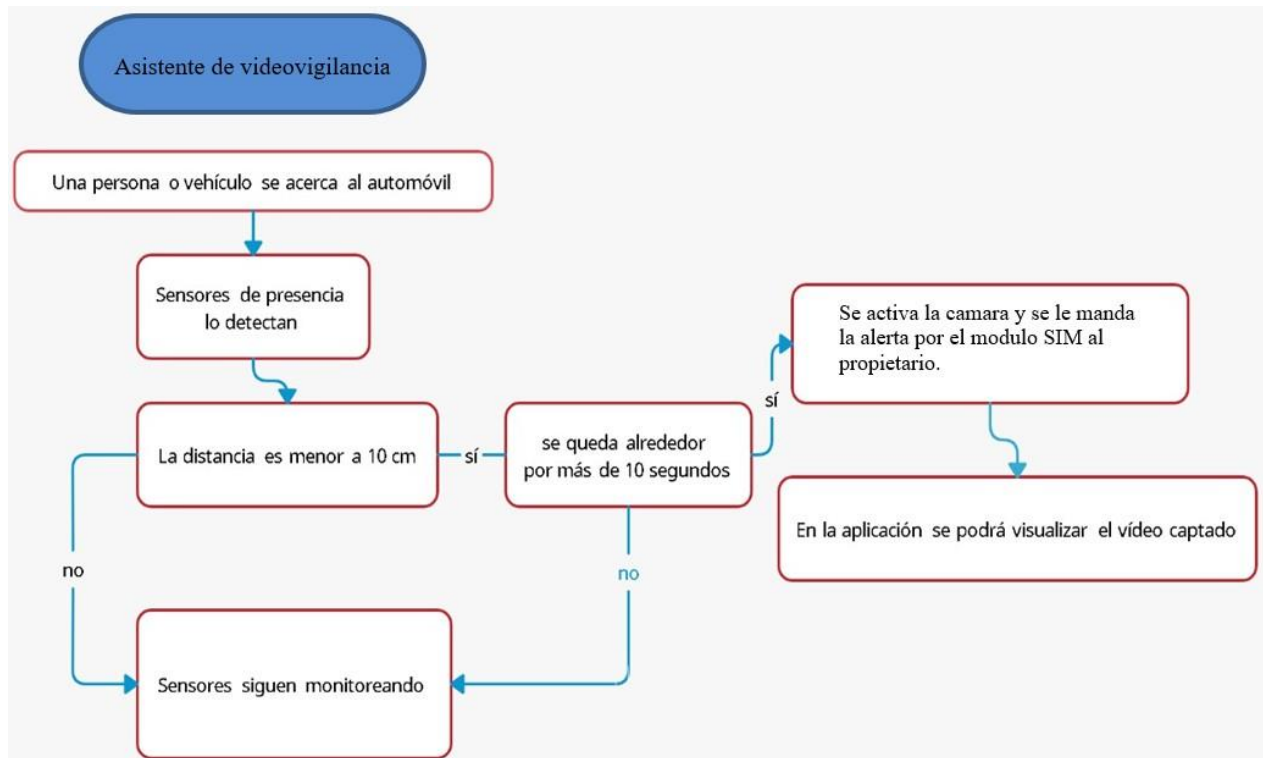


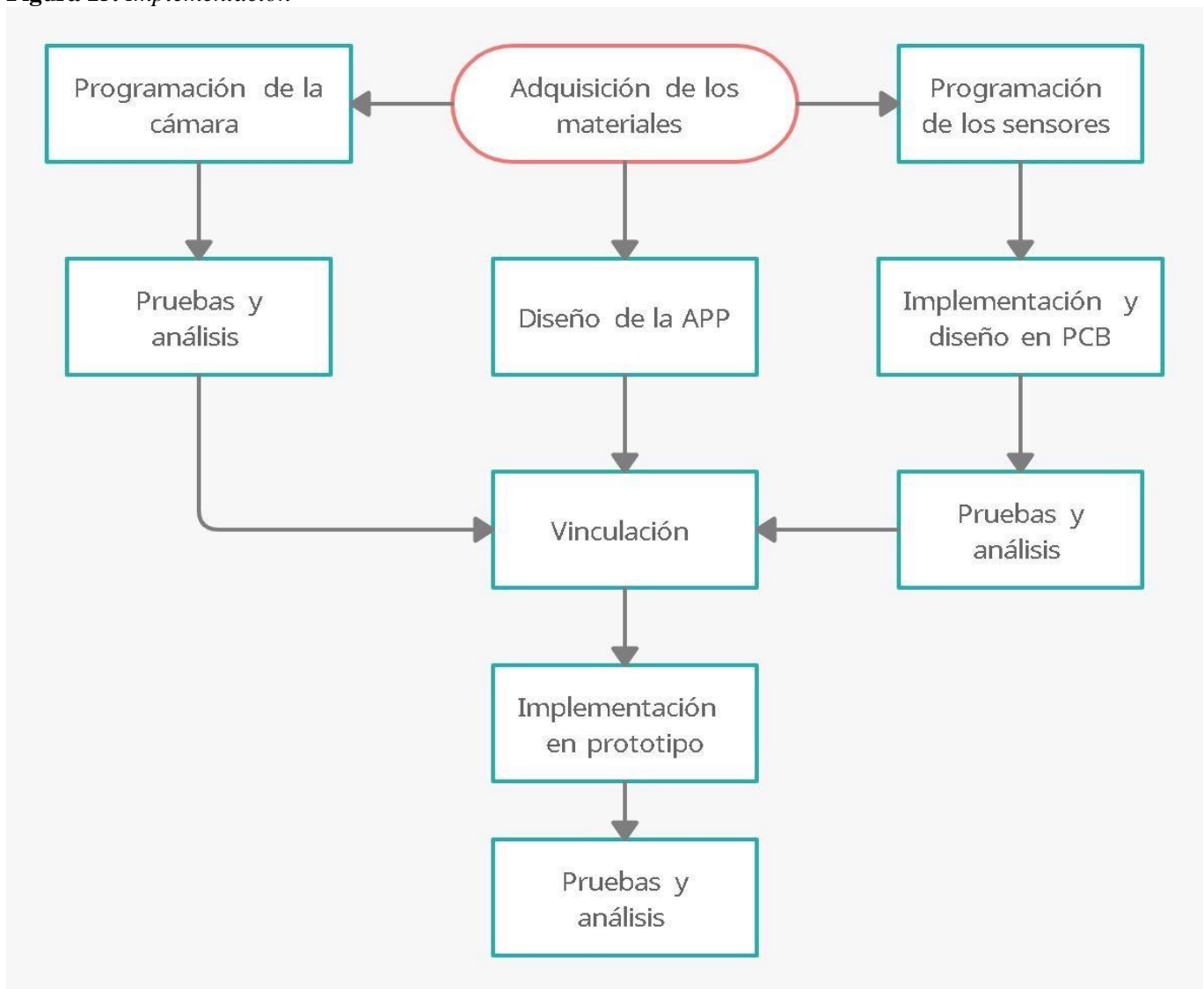
Figura 14. Esquema 2

En este esquema se puede apreciar el funcionamiento que tendrá el prototipo una vez terminado, el funcionamiento es simple pero efectivo. Se puede observar que los sensores estarán todo el tiempo monitoreando, el módulo SIM y las cámaras estarán esperando la señal de los sensores ultrasónicos para poder activarse y comenzar a grabar.

En el siguiente esquema se puede visualizar y comprender los pasos que se llevarán a cabo de una forma general. Este proyecto está dividido en tres partes fundamentalmente: Configuración de los sensores, configuración de la cámara y diseño de la aplicación.

En cada parte se utilizará un lenguaje de programación diferente; Android Studio para la aplicación, Python para la configuración de la cámara y Arduino para los sensores.

Figura 15. *Implementación*





PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

9.5. Selección de una solución adecuada

En este punto de la metodología que mostramos al inicio del punto del Desarrollo, vamos a tratar varios puntos, con lo cual mejoraremos la solución definitiva.

Se conjuntarán diferentes elementos de software y hardware, los cuales facilitarán la vigilancia de un automóvil, de una manera más rápida y accesible, esto se logrará utilizando el teléfono como un medio de enlace, ya que es el dispositivo más usado hoy en día.

El consumo de energía que tendrá el circuito será lo menor posible, esto porque el circuito ira conectado a la batería del automóvil, por lo tanto, si creamos un circuito que consuma demasiada corriente, lo que podría provocar es que en cualquier momento el automóvil deje de funcionar porque se descargó la batería y, por lo tanto, no sería muy viable el proyecto.

Como ya sabemos todo circuito eléctrico en ocasiones llegan a tener fallas, sin embargo, con el trabajo y la información que hemos recaudado durante este tiempo, se tratara de hacer que estas fallas sean mínimas, y en caso de que llegara a suceder una falla, esta no deberá de ser tan grave.

Como ya lo mencionamos en el punto anterior, si llegase a tener el circuito una falla, ya sea por el uso diario o el deterioró de una pieza, esta no será tan cara, porque vamos a construir el circuito con piezas electrónicas que no sean demasiado caras, con esto se logra que el asistente sea más accesible para el público en general.

El conjunto de todos los elementos de nuestro asistente de videovigilancia no tendrá que ser tan complejo, es decir para cuando el usuario presente fallas en alguna pieza del circuito ya sea por los motivos que anteriormente mencionamos, no sea tan difícil de arreglarlo por él mismo o que incluso en caso de no tener ningún conocimiento en el tema, este lo lleve con alguna persona que tenga alguna noción sobre ello, y no le sea tan complicado de entender el funcionamiento.

Se hará una aplicación móvil, porque la mayoría de las personas cuentan con un celular. El propósito de esto es cumplir con uno de nuestros objetivos, el cual es lograr que la vigilancia del auto sea de una manera más rápida y accesible y todo esto se cumplirá a partir del teléfono,



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



el cual se mueve a todas partes con nosotros.

Una vez que el usuario adquiere nuestro asistente de videovigilancia, el usuario tendrá que proporcionar su número de celular, para poder enviar las alertas de seguridad por parte de un módulo SIM.

La aplicación tendrá una base de datos, por lo tanto, para ingresar a la aplicación se tendrá que crear un usuario, esto lo haremos para tener una mayor seguridad de tus datos y de vigilancia de tu auto, que solamente el usuario se entere del estado de su automóvil.

Se hará una vinculación entre la cámara de vigilancia y la aplicación móvil para el envío y recepción de imágenes, así como de los videos, esto mediante la base de datos que se creará, entonces cobra sentido el hacer una aplicación donde se inicie sesión, ya que de esa manera nadie tendrá acceso a los videos e imágenes de un tercero. También el usuario tendrá la opción de descargar la imagen o el video si es que lo desea. Así mismo toda imagen o video tomado quedará guardado en una microSD o memoria USB.



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



A continuación, se muestra un mapa conceptual de los problemas de otros circuitos y las ventajas del nuestro:

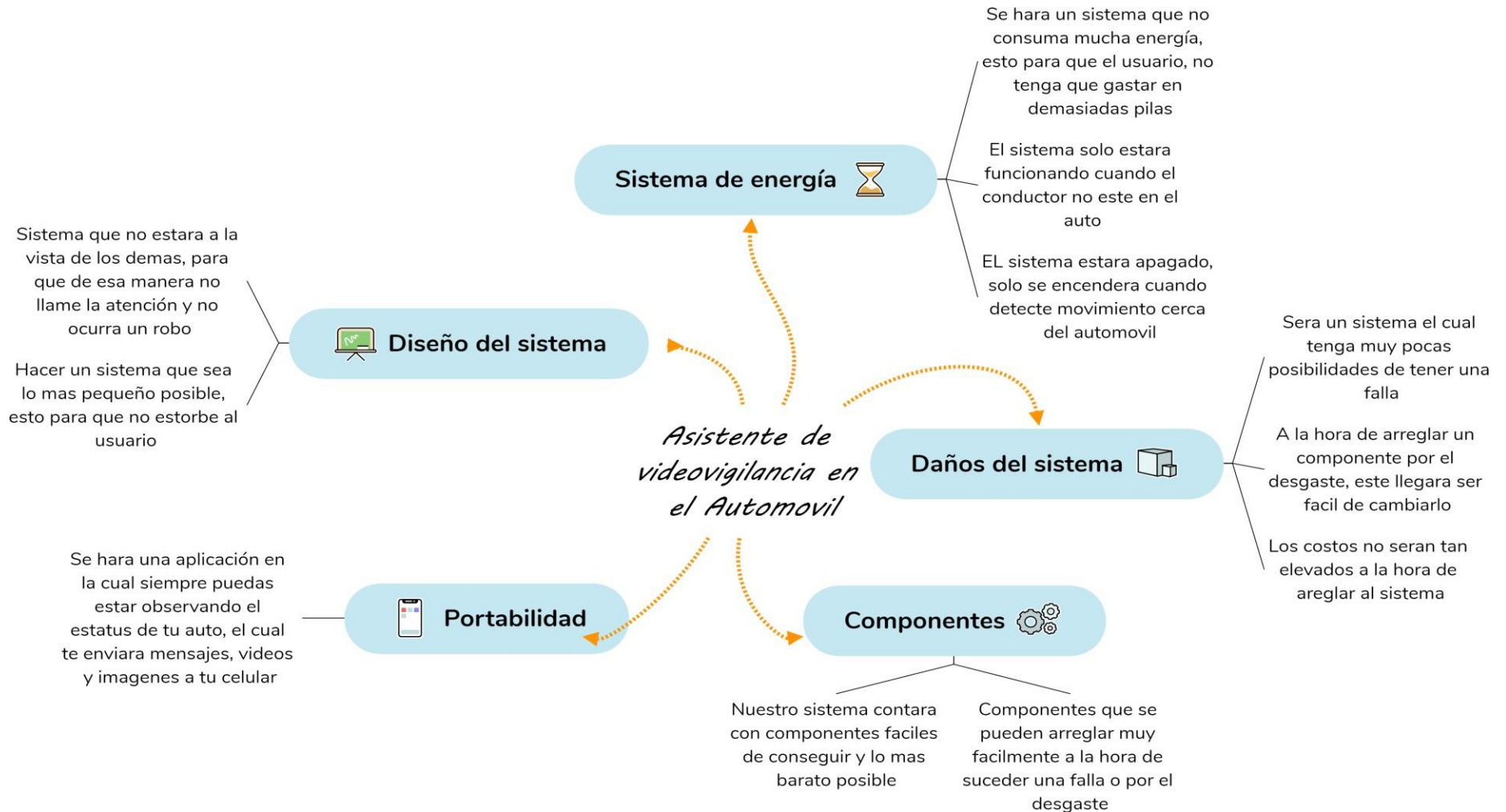


Figura 16. Mapa conceptual, Ventajas de nuestro circuito



9.6. Diseño detallado del sistema

Para diseñar nuestro circuito, recurrimos a cuatro programas:

- ✚ Arduino nano: Programación de los sensores, Modulo Sim y otros componentes (LED, Buzzer, etc.).
- ✚ Android Studio, Programación de la aplicación, con lenguaje Kotlin.
- ✚ Raspberry: Programación de la cámara, como también la transferencia de imágenes y videos a la aplicación.
- ✚ Proteus: Creación de la tabla PCB, donde van todos los componentes conectados.

A continuación, se mostrará el proceso, como también los códigos de cada aplicación:

9.6.1. Arduino

Para poder cumplir con los objetivos descritos previamente, se implementarán los siguientes componentes:

- ✚ Sensor ultrasónico: Es un sensor que tiene la capacidad de detectar obstáculos enfrente de él, tiene un alcance de 4.5 m de distancia, lo cual es ideal para detectar a cualquier persona o vehículo que se acerque. Su funcionamiento es simple pero eficaz. El sensor lanza una onda ultrasónica denominada TRIGGER y ésta “rebota” cuando se encuentra a un obstáculo enfrente, el sensor se encarga de registrar el tiempo que tarda la onda en volver – pin denominado ECHO - y de esta forma se puede calcular la distancia al objeto detectado.
- ✚ En nuestro caso se configuraron los sensores ultrasónicos para alertar al usuario cuando la distancia detectada sea menor a 10 cm. Mientras esta sea mayor a 10 cm no habrá ningún problema.
- ✚ La precisión de este sensor es del 98-99%, es decir que tiene un error del 1-2%. Esta precisión está detallada en las pruebas documentadas.
- ✚ Módulo SIM 800L.- Módulo que se caracteriza por utilizar señal GSM (Global System Communication), su proposito es mandar mensajes con la ayuda de un chip.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Su función en el prototipo es simple: Este módulo será activado únicamente por los sensores ultrasónicos cuando éstos hayan detectado que una persona o vehículo permaneció por más de 10 segundos. En ese momento se activa el módulo SIM, mandando un mensaje de alerta al usuario y de esta forma saber que se ha detectado actividad inusual cerca de nuestro vehículo. Para que este módulo pueda funcionar correctamente únicamente necesita tener un chip con crédito para poder enviar la cantidad de mensajes necesarios.

Gracias a que este módulo opera con la tecnología GSM, puede mandar mensaje a cualquier dispositivo dentro de una zona con conexión a red, es decir, que puede incluso mandar mensajes a un dispositivo fuera del país. La calidad de la señal depende del chip (TELCEL, ATT, UNEFON, etc) que se le ponga.

- ✚ Leds y Buzzer: Servirán como evidencia visual y auditiva para poder comprender de una mejor forma el funcionamiento del prototipo, de igual forma el buzzer será activado por el sensor ultrasónico al detectar una distancia menor a 10 cm. Con ayuda del buzzer se le mandará una advertencia a cualquiera que llegue a acercarse a nuestro vehículo.

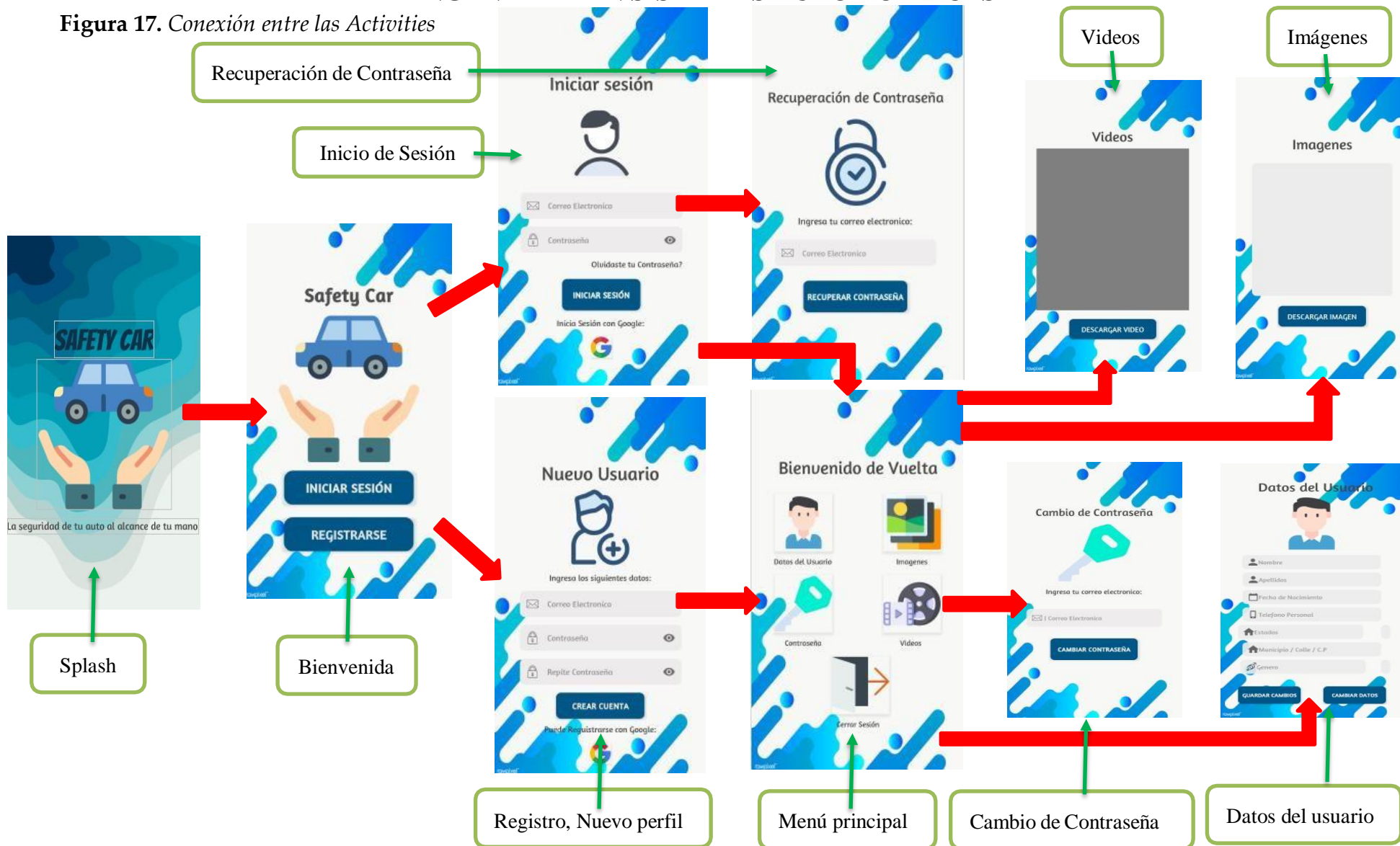
9.6.2. Android Studio

La aplicación que hicimos cuenta con un inicio de sesión, esto para darle más seguridad al usuario, de esa manera, solamente él puede ver las imágenes como también los videos. En la siguiente página se muestra cómo es que se conectan todas las activities entre ellas.



PROYECTO INGENIERÍA EN SISTEMAS AUTOMOTRICES

Figura 17. Conexión entre las Activities





INGENIERÍA EN SISTEMAS AUTOMOTRICES

A continuación, se explica la imagen anterior, para que se entienda como es la estructura de la aplicación y cual es la función de las diferentes ventanas:

- ✚ En la activity splash, es simplemente una activity de diseño, solo sirve para mostrarle al usuario el nombre de la aplicación como también un eslogan, se podría decir que es una presentación de nuestro proyecto.
- ✚ En la activity Bienvenida es donde se muestra el nombre de la aplicación y la imagen, pero muestra dos botones, uno donde es iniciar sesión, y el otro el cual es registrarte.
- ✚ Si le das clic en el botón de registro, este te mandara a la activity, Nuevo usuario, que es donde deberás ingresar tu correo electrónico y crear una contraseña, o puedes registrarte con tu cuenta de Google (Gmail). Si el usuario da clic en el botón de crear cuenta, pero no ha llenado los cuadros de texto este se le enviara un mensaje diciéndole que debe de llenar todos los cuadros de texto.
 - ✓ De la misma manera, si el usuario no ingresa un correo valido no se podrá crear tu perfil, y lo mismo sucede si ingresas una contraseña menor de 6 caracteres o que no se parezcan las contraseñas que ingresaste.
 - ✓ Los mensajes que se mencionan son los siguientes:



Figura 18. Mensajes de Error Activity Registro

- ✚ Si después de ingresar correctamente todos los datos que se te piden, te saldrá un mensaje diciendo que se está cargando tu perfil y una vez que se cargue, te enviará a la activity "menú principal".



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



Figura 19. Mensaje: Cargando Perfil

Cuando ingreses tu correo, este quedara registrado no en la aplicación, este quedara registrado en la base de datos llamada Firebase, la cual te permite guardar los datos de los usuarios registrados, como también imágenes, videos, etc.

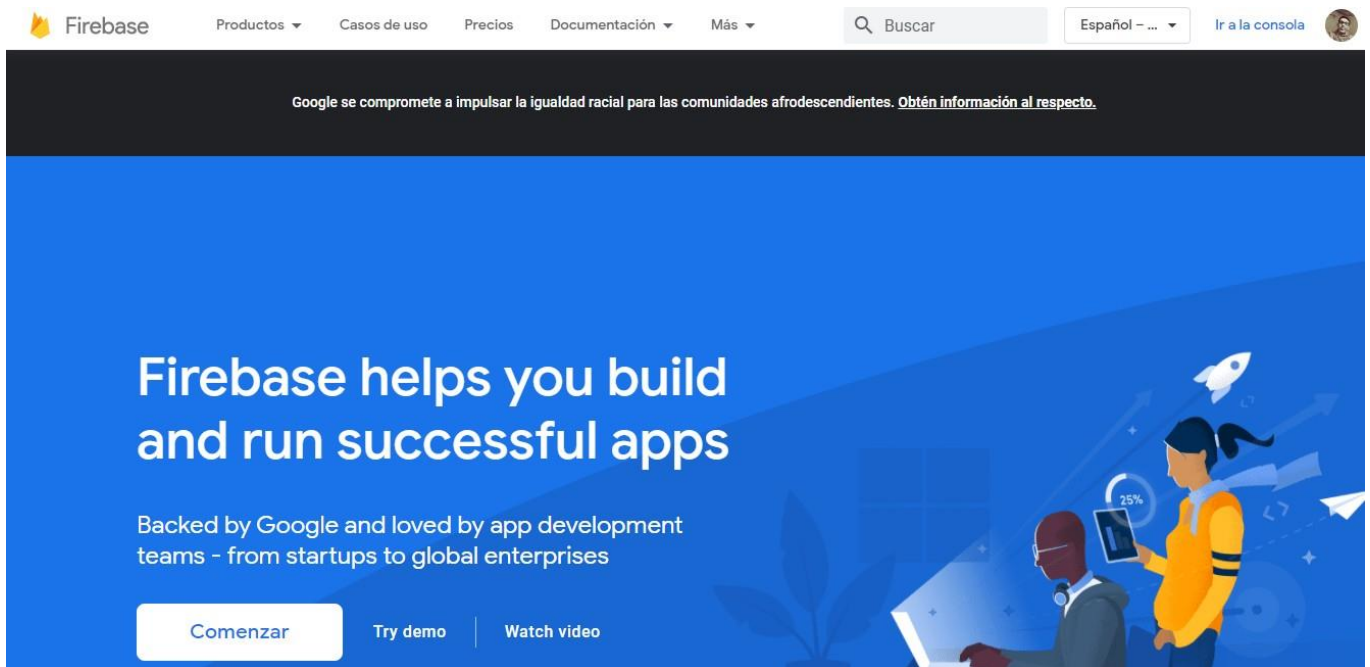


Figura 20. Menú de Firebase

Para crear tu base de datos de Firebase, se hace lo siguiente:

Primero le darás clic en donde dice “comenzar”, una vez hecho esto te aparecerá lo



siguiente:

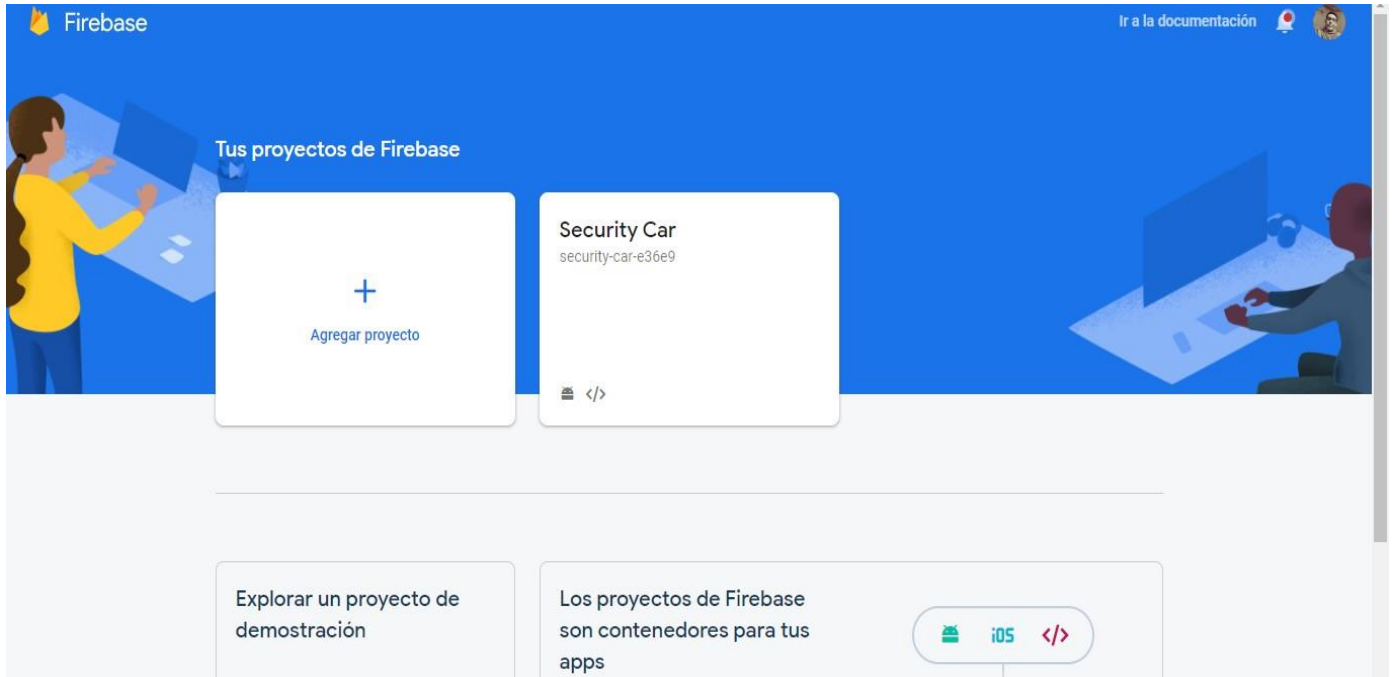


Figura 21. Creación de proyecto.

En esta parte le darás clic en Agregar nuevo proyecto. Después de eso te aparecerán las siguientes ventanas:

- ✚ Ingresar nombre de tu proyecto
- ✚ Después de lo anterior se te explicara que es Google analytics, que es básicamente como funciona Firebase.
- ✚ En la siguiente ventana configuras analytics, escoges la opción “default” para que funcione correctamente Firebase, y le das en crear proyecto, una vez echo eso, se empezara a cargar tu proyecto.

Después de eso te mandara a la siguiente ventana:

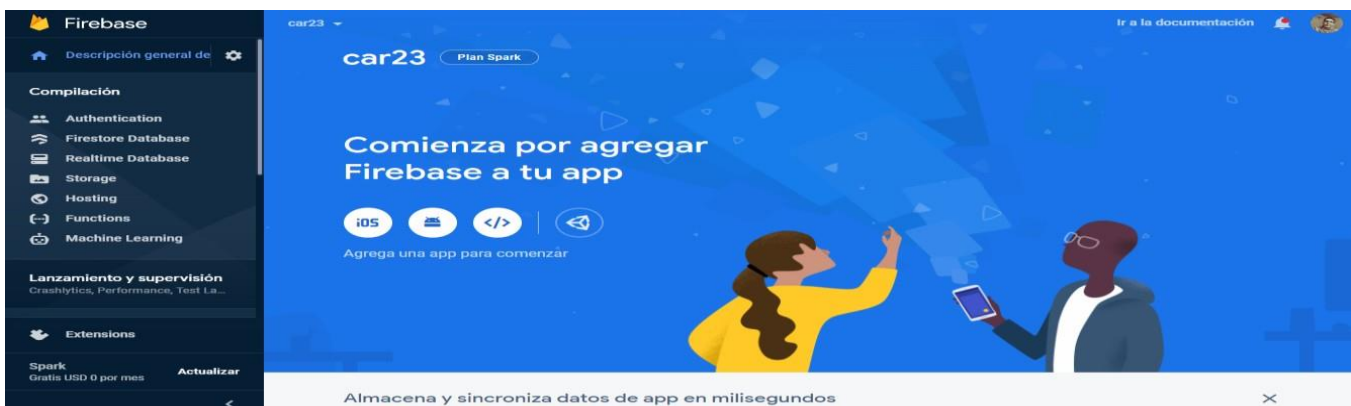


Figura 22. Pantalla principal de tu proyecto



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Lo primero que se hará es agregar tu app, para eso le darás clic en el botón de Android, una vez echo eso, harás todo lo que te pida firebase, este te ira explicando lo que debes de agregar en tu aplicación de Android, como también que es lo que requiere firebase para registrar tu aplicación.

Una vez hecho todo lo anterior, te saldrá la siguiente ventana:

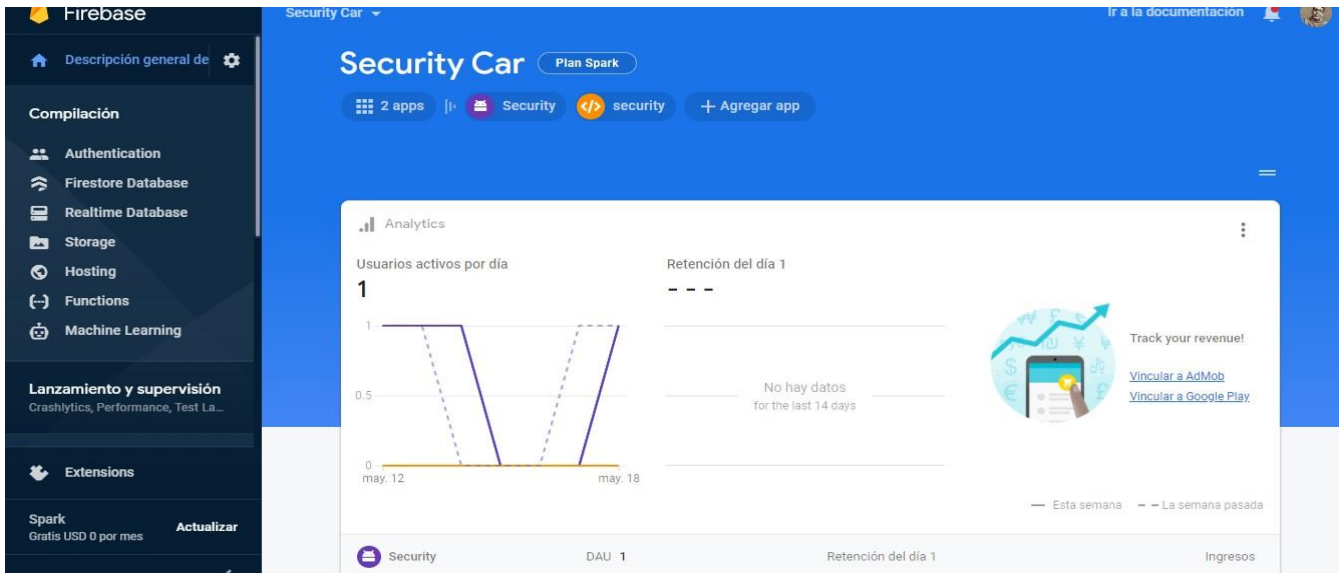


Figura 23. Menú de aplicación registrada

En esta podrás ver los usuarios que se registran en tu aplicación como también las opciones que tienes para agregar a tu aplicación.

Para que tu aplicación tenga habilitada la opción de registro tendrás que ir a la pestaña que dice "Authentication", en ella te aparecerán lo proveedores que se pueden habilitar y registrar, en este caso utilizaremos el correo electrónico y Google, entonces para habilitar estas opciones se hace lo siguiente:

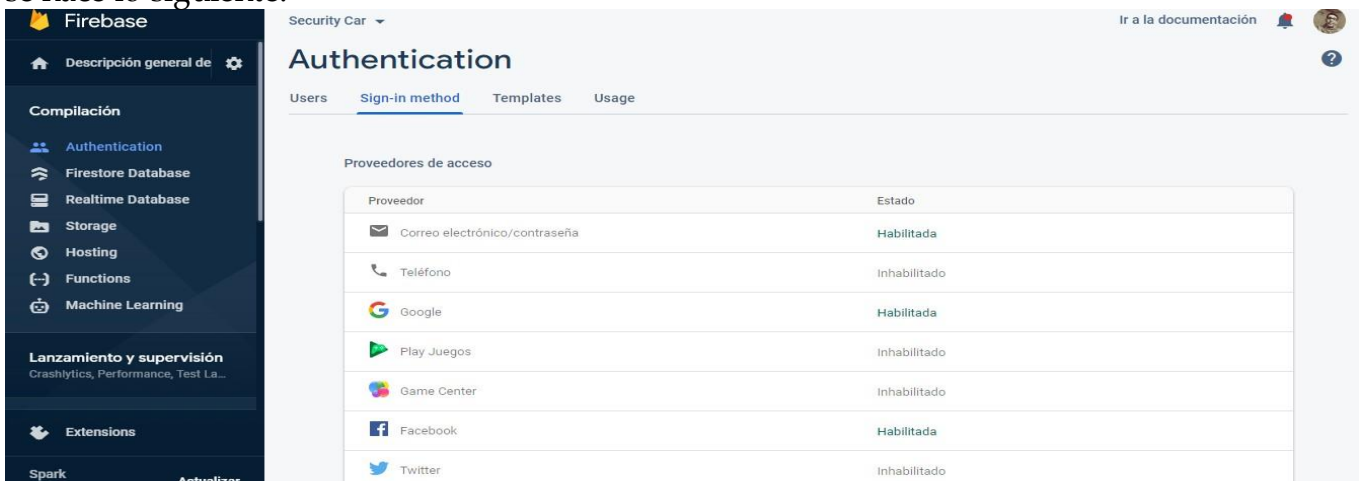


Figura 24. Menú de aplicación registrada



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Una vez que ingresas a la pestaña de Authentication, te aparecen todos los proveedores, le das clic al proveedor que quieres habilitar y te saldrá lo siguiente:



Figura 25. Habilitar proveedor

En este caso ya está habilitada, sin embargo, te aparecerá inhabilitada, solo le darás en habilitar y le das clic en guardar, y estará habilitado tu proveedor. Lo mismo se hará para el proveedor de Google.

Para activar la pestaña de Firestore Database y Storage, lo harás desde Android studio, todo eso se explicará en los códigos más adelante.

- ✚ Básicamente la función de Firestore Database, es guardar datos del usuario, como nombre, teléfono, correo electrónico, fecha de Nacimiento, etc.
- ✚ Por su parte Storage es para guardar imágenes, archivos, videos, etc.

Entonces una vez que ya habilitaste el proveedor y te registraste en tu aplicación, te aparecerá la pestaña authentication, de la siguiente manera:

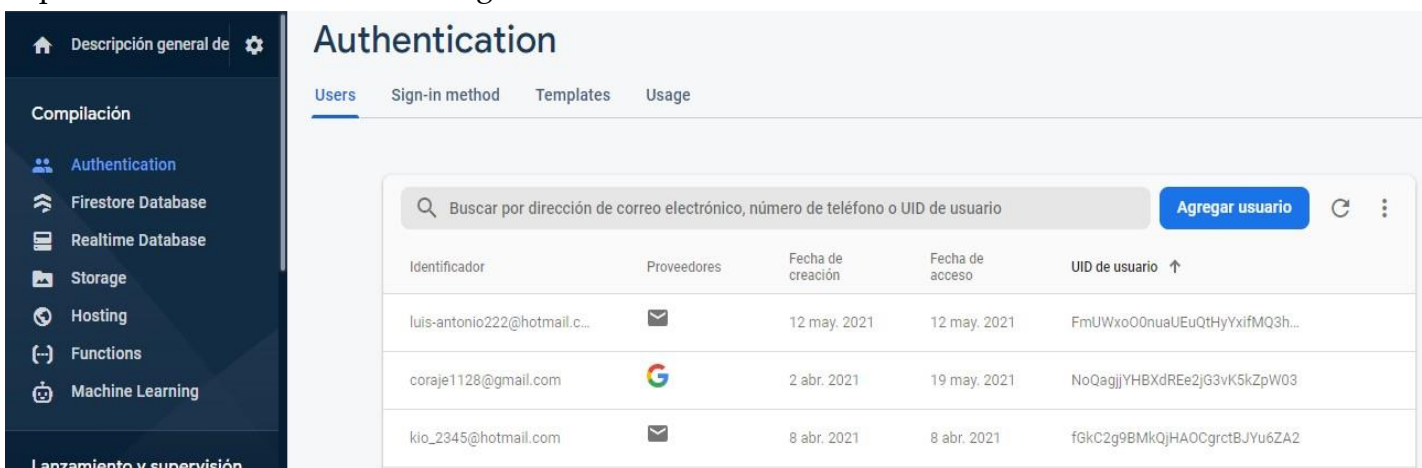


Figura 26. Usuarios Registrados



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Como se puede observar, tenemos tres usuarios registrados en nuestra aplicación, dos usuarios por correo como Outlook, y otro mediante Google. Desde esta página, podemos eliminar los usuarios, inhabilitarlo o cambiar contraseña, sin embargo, esto no se debe de hacer.

- Después de lo anterior, este nos mandara a la activity Menú principal, en la cual como se observa, se tiene 5 botones, uno es donde están los datos del usuario, en este solo agregaremos los datos del dueño, como nombre, fecha de nacimiento, ciudad, genero, teléfono, en caso de no llenar todas las casillas te saldrá una imagen donde se te pide llenarlas para poder continuar, una vez que lo hayas echo, estos datos quedaran guardados en Firebase, en la pestaña Firebase Database, como se muestra en la siguiente imagen:

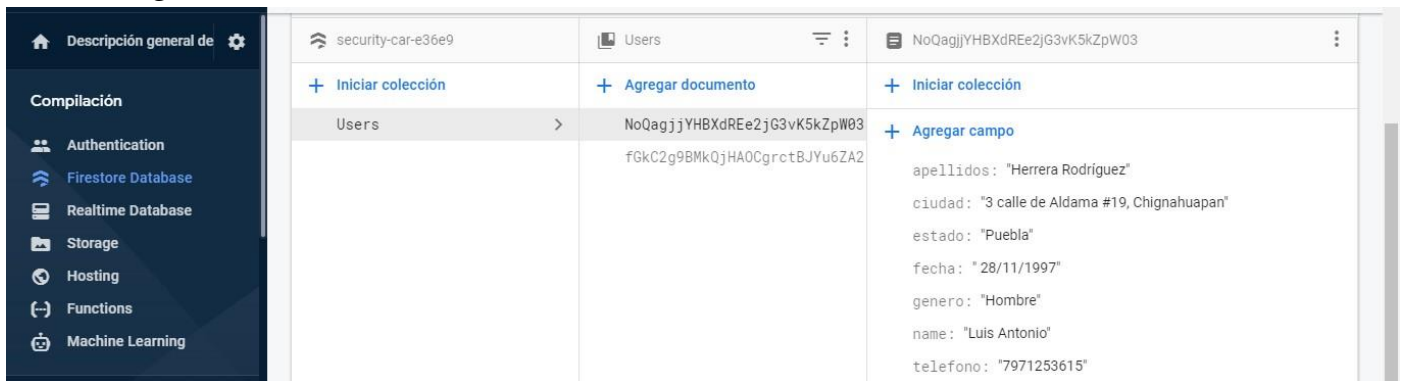


Figura 27. Datos del usuario registrado

Como se puede ver en la Figura 31, muestra los datos de la persona registrada, se puede ver que hay en total 2 personas que proporcionaron sus datos.

- En la activity Videos, solamente se mostrará el video que envíe la raspberry, este tema se mostrará en el punto 9.6.3. que corresponde a Raspberry.
- Lo mismo sucede en la activity imágenes, en esta activity se mostrará las imágenes enviadas por las Raspberry.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

- En la actividad cambio de contraseña, el usuario tendrá que ingresar su correo electrónico en caso de querer cambiar su contraseña, en caso de ingresar un correo que no esté registrado en la aplicación, este te mandará un mensaje de error, pero si ingresas un correo registrado, tendrás que ir a la bandeja de tu correo electrónico y te aparecerá el siguiente correo:



Figura 28. Correo de recuperación

Te mandará un link, el cual servirá para poder cambiar la contraseña, cuando le das clic te aparecerá la siguiente ventana:

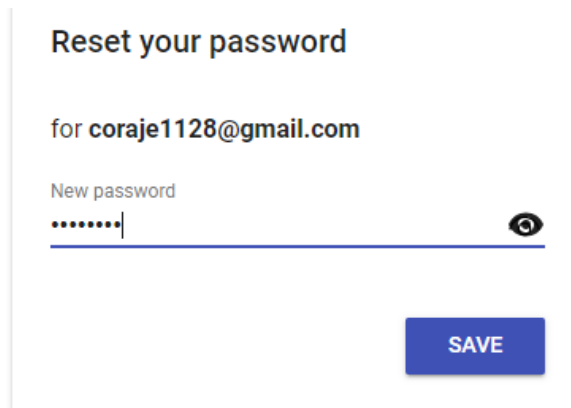


Figura 29. Cambio de contraseña

De esta manera solo ingresas tu nueva contraseña y listo.

- El cuarto botón, el cual es, cerrar sesión, cuando le das clic, este te mandara un mensaje preguntando si quieres cerrar sesión, en caso de que no este mensaje se omitirá, pero en caso de que no, este te mandara a la actividad Bienvenida.



Figura 30. Mensaje de cerrar sesión

- Cuando estas en la activity Bienvenida, y le das clic en iniciar sesión, este te mandara a la activity Registro, en el cual debes de ingresar tu correo electrónico y contraseña que registraste en Firebase, en caso de no tener ninguna cuenta registrada, o ingresar una contraseña incorrecta, esta te mandara un mensaje de error, de igual manera puedes iniciar tu sesión con tu cuenta de Google que ya registraste.



Figura 31. Mensajes de error, Activity Registro

- La activity Recuperación de contraseña, es lo mismo que la activity cambio de contraseña, te llega el mismo correo, sin embargo, para acceder a esta activity, le debes de dar clic en donde dice “¿has olvidado tu contraseña?”, y una vez que das clic te dirige a esta activity.

9.6.3. Raspberry

Lo que se hizo de primera instancia fue comprar la cámara de la raspberry, la cual es una cámara que tiene un lente angular grande, lo que ayuda este lente angular es tomar una foto mas amplia, logrando captar más objetos en una sola foto.

Esta se coloco en el puerto que tiene la raspberry, que funciona para colocar este tipo de componentes.

Para la comunicación que tiene las raspberry con firebase se hace de la siguiente manera.

Para poder enviar las imágenes que toma la cámara, se hace lo siguiente:

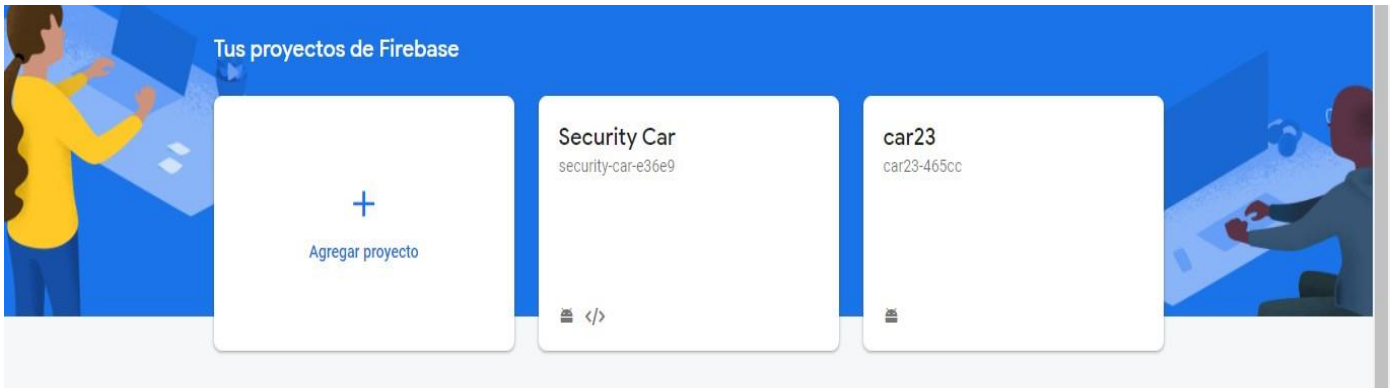


Figura 32. Proyectos en Firebase

Una vez dentro de la pagina principal de firebase y de tus proyectos, daras clic al proyecto que quieres, una vez dentro de tu proyecto te aparecerá la siguiente imagen:



Figura 33. Apps disponibles en firebase

En ella escogerás en agregar app, y escogerás la opción de “web”, una vez que le das clic en esta opción te saldrá lo siguiente:



Figura 34. Firebase en aplicación web

Dentro de ella solo colocas el nombre de tu agrado para tu proyecto, una vez echo eso le das clic en registrar app, y te saldrá lo siguiente:



Copla y pega estas secuencias de comandos en la parte inferior de la etiqueta <body> antes de usar cualquier servicio de Firebase:

```
<!-- The core Firebase JS SDK is always required and must be listed first -->
<script src="https://www.gstatic.com/firebasejs/8.6.2/firebase-app.js"></script>

<!-- TODO: Add SDKs for Firebase products that you want to use
https://firebase.google.com/docs/web/setup#available-libraries -->
<script src="https://www.gstatic.com/firebasejs/8.6.2/firebase-analytics.js"></script>

<script>
  // Your web app's Firebase configuration
  // For Firebase JS SDK v7.20.0 and later, measurementId is optional
  var firebaseConfig = {
    apiKey: "AIzaSyCcoW3SnGbbjDgdxYOAEF9qphVMu5BrSU",
    authDomain: "security-car-e36e9.firebaseio.com",
    databaseURL: "https://security-car-e36e9-default-rtdb.firebaseio.com",
    projectId: "security-car-e36e9",
    storageBucket: "security-car-e36e9.appspot.com",
    messagingSenderId: "1003061248625",
    appId: "1:1003061248625:web:2a062043dc0f54b0cb51d0",
    measurementId: "G-JTE645Z678"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
  firebase.analytics();
</script>
```

Figura 35. Código para agregar a código de Python

En la figura 35, se muestra un código, el cual copiaras, y pegaras en tu código de Python. Una vez hecho esto, estará listo para enviar las imágenes de raspberry a firebase, y de firebase a tu aplicación.

En la siguiente imagen se muestra como queda guardada la imagen y el video en firabase:

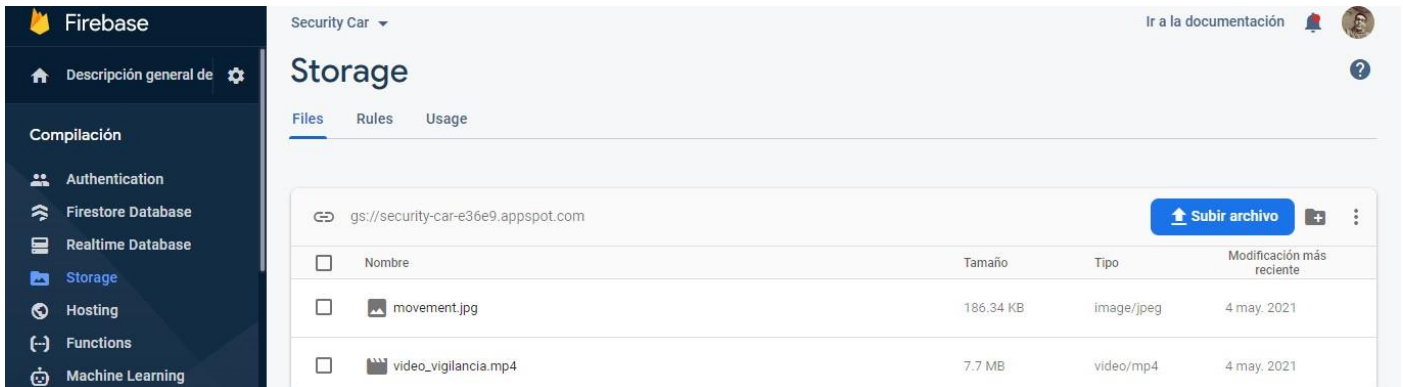


Figura 36. Imagen y video guardado en firebase

Ahora, para que funcione la raspberry, tiene que estar conectada a una fuente de poder de 5v y 3 Amp, en este caso optamos por una batería para motocicleta de 12v y 12 Amp, esto porque queremos que el asistente funcione durante demasiado tiempo, y así el usuario no se preocupe de que la batería se descargue rápido por otra parte se quiso asemejar el funcionamiento y las condiciones que se presentan en un automóvil.

Como ya se mencionó, la raspberry soporta hasta 5v, por lo tanto, para que funcione

INGENIERÍA EN SISTEMAS AUTOMOTRICES

adecuadamente con la batería de 12v, decidimos comprar un reductor de voltaje, el cual hace que a la salida se tenga el voltaje que uno requiera, en este caso los 5v, pero a su vez que tenga una corriente de salida mayor a 3 Amp ya que eso requiere la raspberry para trabajar adecuadamente, de esa manera la raspberry se energiza correctamente y no corre riesgo alguno.

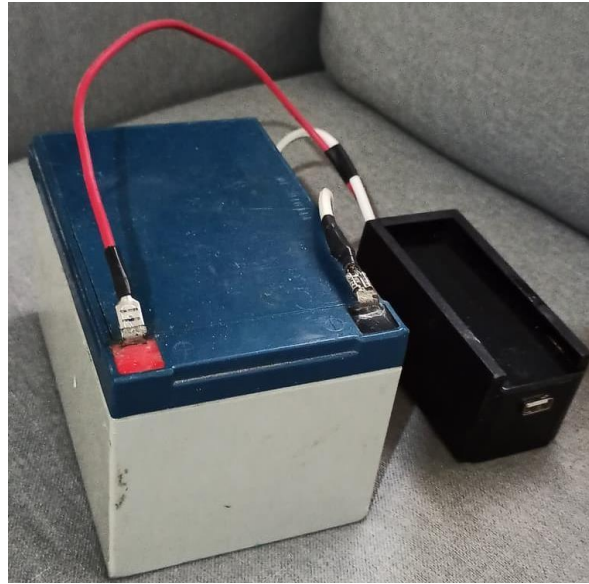


Figura 37. Pila y reductor de voltaje.

9.6.4. PCB

En la siguiente imagen se muestra como quedo el diseño final de la PCB.

En ella se puede observar como quedo montado el Arduino nano, como también las pistas, que es donde saldrán todos los componentes del asistente.

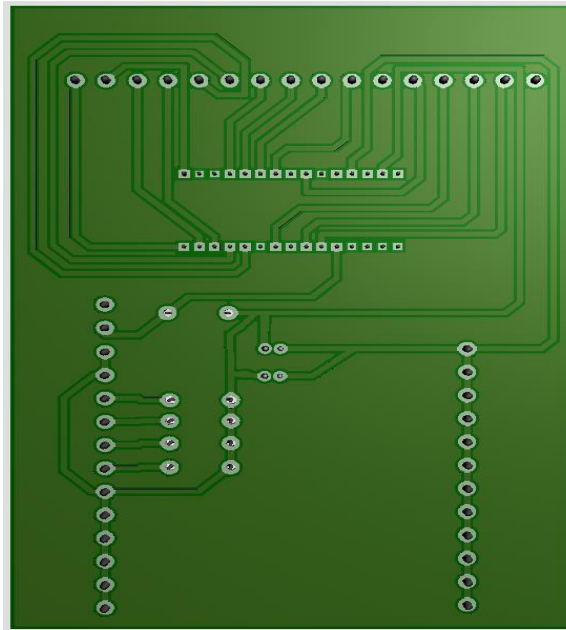


Figura 38. Diseño de la PCB, pistas.

INGENIERÍA EN SISTEMAS AUTOMOTRICES

Se optó por utilizar el Arduino nano porque es pequeño, por lo tanto, resulta menos estorboso a la hora de montarlo, además de que todo nuestro asistente está montado en una caja de madera pequeña, entonces si poníamos otro modelo se reduciría el espacio. Además, su consumo de energía es menor a comparación de otros modelos, por otra parte, cuenta con el número de pines que requerimos.

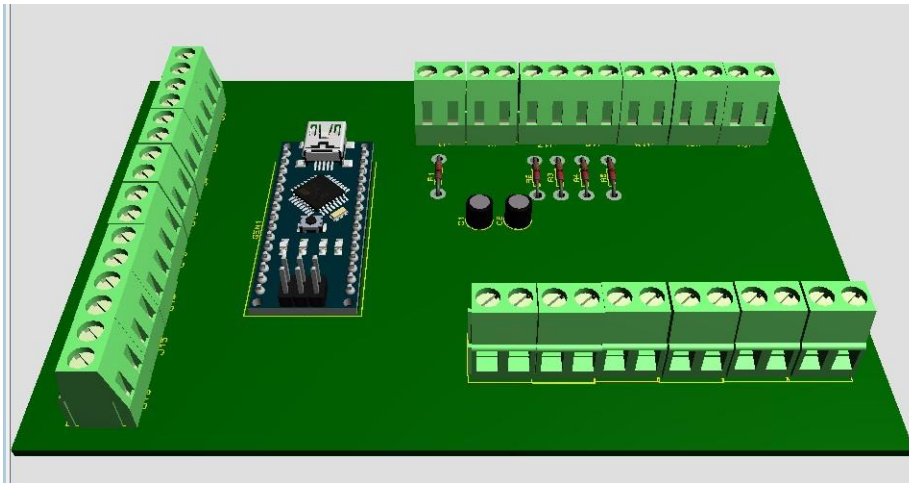


Figura 39. *Diseño de la PCB, Lateral.*

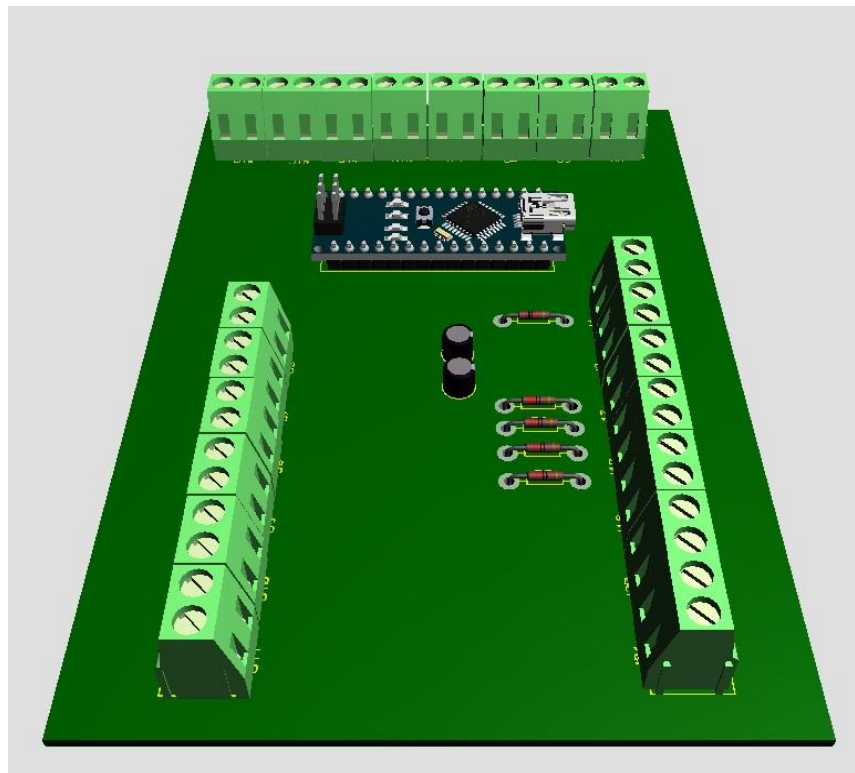


Figura 40. *Diseño de la PCB, frontal.*



9.7. Construcción de prototipo

Una vez hecho todo lo anterior, lo cual fue programar cada uno de los subsistemas que cuenta nuestro asistente (Arduino, Android Studio, Raspberry), fue implementar todos los sensores, la raspberry, el Arduino nano, la placa pcb y la cámara en el auto a escala, el cual simulara un automóvil real. En pocas palabras este auto simplemente es un prototipo, de como quedaría nuestro asistente, si es que este llegara a montarse en un automóvil real.

Lo primero que hicimos, fue hacer todas las modificaciones en el auto de juguete, como por ejemplo hacer las bases en donde irían montados los sensores, como también los servomotores, los agujeros en donde irían montados los leds, botones, buzzers, cámara, etc. Como también algunas modificaciones para que el auto se viera más presentable. Todo esto se puede observar en las siguientes imágenes:



Figura 41. *Auto sin modificaciones*



INGENIERÍA EN SISTEMAS AUTOMOTRICES



Figura 42. *Modificaciones al auto 1.*



Figura 43. *Modificaciones al auto 2.*

INGENIERÍA EN SISTEMAS AUTOMOTRICES



Figura 44. Base del sensor ultrasónico y cámara.

Una vez echo lo anterior, comenzamos con la elaboración de la tabla de PCB, como ya se mostró anteriormente, hicimos el diseño de la PCB, en PROTEUS, una vez que teníamos este diseño, imprimimos este diseño en una hoja especial, el cual tenemos que imprimir con una máquina de lasser, esto para que, a la hora de planchar el circuito en la placa de cobre, este se transfiera adecuadamente y no se llegue a despegar.

Una vez hecho esto, tuvimos que colocarla en cloruro férrico, para que eliminara todo el cobre, y solo quedaran las pistas, en las cuales se soldaran todos los componentes del circuito.

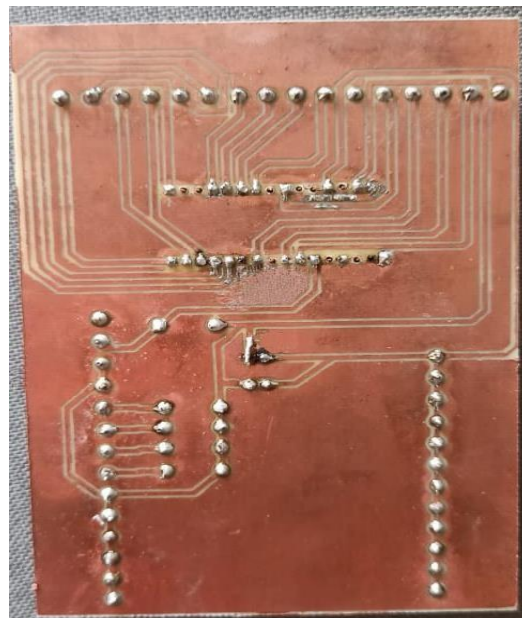


Figura 45. Placa PCB.

INGENIERÍA EN SISTEMAS AUTOMOTRICES

Una vez hecho esto, comenzamos a soldar todos los componentes que utilizamos, como fueron los sensores, Leds, buzzer, botones, servomotres, el modulo sim y el Arduino nano. Todo esto lo hicimos con cable de un calibre AWG 22, esto para que no se dañaran tan rápidamente, y también para que estos no se llegasen a quemar por la corriente que van a tener que soportar, que en si es mínima.

Esta placa la colocamos en una caja de madera pequeña, la cual ira en la parte del asiento, esto para cumplir con el objetivo de hacer un asistente pequeño, y no estorbe a la hora de montarlo en un auto real.

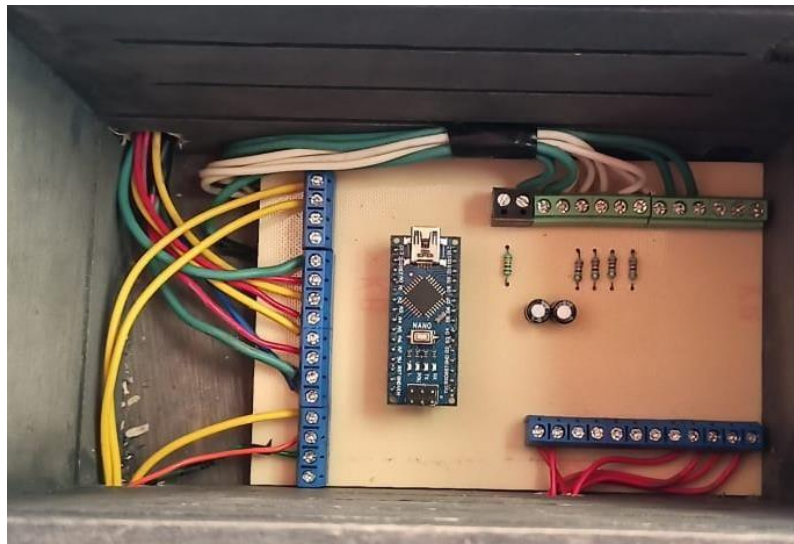


Figura 46. Placa PCB instalada en la caja de madera

Una vez que se conectaron todos los componentes en la placa PCB, comenzamos a instalar la raspberry, en la misma caja de madera. De igual manera, instalamos la cámara en el espacio que definimos antes, como también todos los sensores, leds y módulos sim, en sus respectivos espacios, quedando de la siguiente manera:

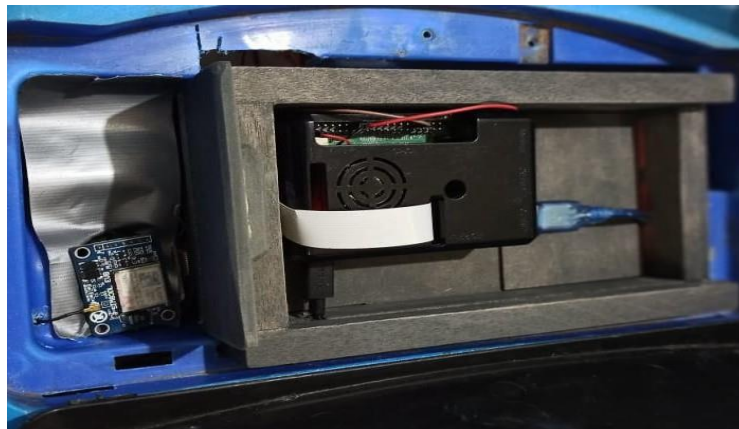


Figura 47. Placa PCB, Arduino, Raspberry y Modulo Sim 1.

PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES



Figura 48. Placa PCB, Arduino, Raspberry y Modulo Sim 2.

Leds:

- ✚ Rojo: representa cuando se envió el mensaje, como también cuando un automóvil o persona se quedó más de diez segundos dentro de la zona más cercana al automóvil (10cm).
- ✚ Amarillo: este parpadeara hasta que los sensores dejen de notar alguna presencia menor o igual a 15 cm.
- ✚ Verde: Este se encenderá mientras los sensores no noten alguna presencia, es decir que la distancia entre el objeto y el auto sea mayor a los 15 cm.

El botón rojo, es aquel que modificara el tiempo que durara para enviar el mensaje de alerta al usuario, para saber cuánto tiempo se coloca, se utilizó un led RGB:

- ✚ Si el led RGB se ilumina en un tono azul, significa 10 segundos para enviar el mensaje.
- ✚ Si el led RGB se ilumina en un tono morado, significa que enviara el mensaje después de 15 segundos.
- ✚ Si el led RGB se ilumina en un tono color rojo, significa que enviara el mensaje después de 20 segundos.



Figura 49. *Tablero del Automóvil, Boton de usuario y leds.*



Figura 50. *Sensores, servomotor y cámara instalados.*



Figura 51. *Sensor de agua instalado.*

Todo nuestro asistente estará alimentado por una batería de moto, la cual es de 12v y 12000Amp-H, en la cual saldrán dos cables y llegarán al reductor de voltaje, esto para que no queme a la raspberry, ya que esta batería alimentara a la raspberry, y la raspberry alimentara al Arduino nano. El reductor de voltaje, tuvimos que ajustarlo a 5v, ya que es el voltaje que soporta la raspberry y tendrá una salida de corriente de 4 Ampers.



Figura 52. *Reductor de voltaje y pila de moto 1.*



Figura 53. Reductor de voltaje y batería de moto 2.



Figura 54. batería montada en el automóvil.

Por ultimo, se muestra una foto tomada por la cámara, la cual como se puede ver es de buena calidad, y logra captar varios objetos en una sola foto.



INGENIERÍA EN SISTEMAS AUTOMOTRICES



Figura 55. Foto tomada por la cámara

Ahora, al final de las pruebas que realizamos, las cuales se pueden observar en los videos, ya no utilizamos el sensor de lluvia y el sensor de efecto Doppler esto por lo siguiente:

El sensor de efecto Doppler tenía el objetivo de sustituir a los sensores ultrasónicos cuando lloviera, ya que se pensaba que estos detectaban las gotas de lluvia por otra parte el sensor de lluvia era el encargado de detectar cuando estuviera lloviendo y de esa forma activar el Doppler y desactivar los sensores ultrasónicos, pero con las pruebas que se hicieron, se demostró que los ultrasónicos no detectan las gotas de lluvia, por ende, se descartaron los sensores de lluvia y de efecto Doppler. Gracias a que se quitaron los dos sensores, el consumo energético disminuyó 20 mA por parte del sensor de agua y 3 mA por el lado del sensor de efecto Doppler.

9.7.1. Implementación del prototipo

La implementación de este prototipo se realizó únicamente en la parte trasera del automóvil por las siguientes razones:

- ✚ **Recursos Económicos:** La idea es implementar el prototipo en todo el automóvil (parte trasera, delantera y costados) cubriendo todos los lados y tener un monitoreo completo, sin embargo, para la adquisición de los demás sensores y cámaras se requería un gasto económico mayor, el cual superaba el presupuesto estimado para la realización del



INGENIERÍA EN SISTEMAS AUTOMOTRICES

prototipo. Por esta razón se optó por cubrir solamente un lado del automóvil que es la parte trasera y evaluar el funcionamiento del prototipo para que en un futuro pueda ser implementado en los demás lados restantes.

✚ **Pruebas y Análisis:** Una vez implementado el prototipo en la parte trasera se realizaron las pruebas correspondientes para evaluar su funcionamiento.

Los resultados fueron los esperados, el prototipo cumple con los objetivos propuestos de una manera eficiente. Como se mencionó en el punto anterior, el objetivo era evaluar el funcionamiento en la parte trasera.

Finalmente, el prototipo funciona correctamente, por lo tanto, esto sirve como evidencia de que, con los recursos económicos necesarios y con los materiales o dispositivos que ayuden a vincular más de una cámara y que estas trabajen adecuadamente, este prototipo puede ser implementado en los cuatro lados del automóvil.



INGENIERÍA EN SISTEMAS AUTOMOTRICES

9.8. Costo y horas del proyecto

Tabla 6. Costo de los componentes utilizados en el asistente y Horas de trabajo

GASTOS DIRECTOS	CONCEPTO	CANTIDAD	UNIDAD	PRECIO UNITARIO	TOTAL
MATERIALES					
	Sensor Ultrasónico HC-SR04	3	PIEZA	\$ 28.00	\$ 84.00
	Led Rojo	1	PIEZA	\$ 3.00	\$ 3.00
	Led Amarillo	1	PIEZA	\$ 3.00	\$ 3.00
	Led Verde	1	PIEZA	\$ 3.00	\$ 3.00
	Led RGB	1	PIEZA	\$ 8.00	\$ 8.00
	Buzzer Pasivo	1	PIEZA	\$ 5.00	\$ 5.00
	Push Button	1	PIEZA	\$ 7.00	\$ 7.00
	Módulo SIM 800L	1	PIEZA	\$ 260.00	\$ 260.00
	Servo Motor	1	PIEZA	\$ 32.00	\$ 32.00
	Cable	20	METROS	\$ 3.50	\$ 70.00
	Cámara 5MPX	1	PIEZA	\$ 350.00	\$ 350.00
	Raspberry PI 4 8GB RAM	1	PIEZA	\$ 2,600.00	\$ 2,600.00
	Arduino Nano	1	PIEZA	\$ 105.00	\$ 105.00
	Regulador de Voltaje	1	PIEZA	\$ 110.00	\$ 110.00
	Batería 12V - 12Ahr	1	PIEZA	\$ 700.00	\$ 700.00
	Resistencia 220 Ohm	6	PIEZA	\$ 2.00	\$ 12.00
	Resistencia 10K ohm	1	PIEZA	\$ 2.00	\$ 2.00
	Baquelita Cobre 9x13	1	PIEZA	\$ 18.00	\$ 18.00
	Impresión papel couche	1	PIEZA	\$ 14.50	\$ 14.50
	Ácido Férrico	200	Mililitros	\$ 40.00	\$ 40.00
	Bornera cuatro pines	12	PIEZA	\$ 6.00	\$ 72.00
	Carrito Monoplaza	1	PIEZA	DONADO	
MANO DE OBRA					



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
PROYECTO



INGENIERÍA EN SISTEMAS AUTOMOTRICES

	Programación de los sensores	30	HORA	\$ 76.50	\$ 2,295.00
	Programación de la cámara	5	HORA	\$ 76.50	\$ 382.50
	Diseño de la Aplicación	50	HORA	\$ 76.50	\$ 3,825.00
GASTOS INDIRECTOS					
	Transporte	6	PESOS	\$ 25.00	\$ 150.00
	Comidas	6	PESOS	\$ 55.00	\$ 330.00
GASTOS INESPERADOS					\$ 300.00
GASTO TOTAL					\$ 11,781.00
GANANCIA ESPERADA					\$ 6,000.00
TOTAL					\$ 17,781.00



10. Conclusiones

La evidencia que presentamos anteriormente demuestra que cada uno de los objetivos fueron realizados siguiendo un proceso claro y detallado.

Cada sensor fue seleccionado de acuerdo con las necesidades requeridas y probado para su funcionamiento óptimo, con ayuda de Arduino se logró establecer las secuencias lógicas que realiza el asistente; Con ayuda de la Raspberry, Python y la cámara se logró obtener el vídeo captado con una buena resolución y almacenando estos archivos en la aplicación; Finalmente la aplicación se diseñó de una manera atractiva e intuitiva para el usuario.

El prototipo final cumple con las necesidades requeridas y entrega los resultados esperados de una manera adecuada.



11. Apéndice

11.1. Código de Arduino:

```
//Incluir librerías
#include <LowPower.h>
#include <Servo.h>
#include <SoftwareSerial.h> //Librería para utilizar pines digitales como puertos
seriales
SoftwareSerial mySerial(10,11); //TX, RX Creamos un objeto llamado SIM800L y
definimos los pines
Servo servo1;

//Declarando variables
const int timeInterval = 150;
const int TRIGGER3      = A0;
const int Buzzer        = A1;
const int LedYellow     = A2;
const int LedRed_RGB    = A3;
const int LedGreen_RGB  = A4;
const int LedBlue_RGB   = A5;
const int Button        = 2;
const int ECHO3         = 3;
const int ECHO1         = 4;
const int ECHO2         = 5;
const int TRIGGER1      = 6;
const int TRIGGER2      = 7;
const int LedGreen      = 8;
const int LedRed        = 9;
int pins[] = {6,7,8,9};
int analogPins[] = {A0,A1,A2,A3,A4,A5};
//volatile boolean doppler1_actived = false;
//volatile boolean doppler2_actived = false;
//RX - 11
//TX - 10
String mensaje = "";
int dopplerDetection = 0;
int distancia1 = 0;
int distancia2 = 0;
int distancia3 = 0;
int tiempo1 = 0;
int tiempo2 = 0;
int tiempo3 = 0;
int counter = 0;
long timePassed = 0;
volatile int tiempoMSG = 10;

//Configurando pines como entradas o salidas
void setup(){
  Serial.begin(9600); // Iniciamos la comunicación serial con Arduino IDE
```



PROYECTO INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
mySerial.begin(9600);

for(int x = 0; x <= 3; x++){
  pinMode(pins[x], OUTPUT);
}
for(int y = 0; y <= 4; y++){
  pinMode(analogPins[y], OUTPUT);
}
pinMode(Button, INPUT);
pinMode(ECHO1, INPUT);
pinMode(ECHO2, INPUT);
pinMode(ECHO3, INPUT);

servo1.attach(13);
servo1.write(90);

attachInterrupt(digitalPinToInterrupt(2), buttonPressed, FALLING);

analogWrite(LedRed_RGB, 255);
analogWrite(LedGreen_RGB, 255);
analogWrite(LedBlue_RGB, 255);
}

//Empieza el ciclo
void loop(){

  distancia1 = CalcularDistancia_1();
  distancia2 = CalcularDistancia_2();
  distancia3 = CalcularDistancia_3();

  //Posicionando la camara
  if(distancia1 <= 15)
    servo1.write(90);
  else if(distancia2 <= 15)
    servo1.write(110);
  else if(distancia3 <= 15)
    servo1.write(70);
  else
    servo1.write(90);

  //Imprimiendo distancias en el monitor serie
  Serial.println("d1: " + String(distancia1) + "\td2: " + String(distancia2) +
"\td3: " + String(distancia3)); // Mostramos serialmente la distancia en pantalla
  delay(25);

  while( (distancia1<= 15) || (distancia2<=15) || (distancia3<=15)){
    //Buzzer----
    tone(Buzzer, 4000, 400);
    //-----
```



PROYECTO INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
digitalWrite(LedRed, LOW);
digitalWrite(LedGreen, LOW);
digitalWrite(LedYellow, HIGH);
distancia1 = CalcularDistancia_1();
distancia2 = CalcularDistancia_2();
distancia3 = CalcularDistancia_3();
counter++;
delay(1000);
//Si ya pasaron 10 segundos entonces mandamos el mensaje
if(counter >= tiempoMSG){
    noTone(Buzzer);
    digitalWrite(LedRed, HIGH);
    digitalWrite(LedGreen, LOW);
    digitalWrite(LedYellow, LOW);
    counter = 0;
    Serial.println("Sending Message...");
    delay(1000);
    mySerial.println("AT"); //Once the handshake test is successful, it will back
to OK
    updateSerial();
    mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
    updateSerial();
    mySerial.println("AT+CMGS=\""+525544776477+"\"); //change ZZ with country code
and xxxxxxxxxxxx with phone number to sms
    updateSerial();
    mySerial.println("ALERTA! SE HA DETECTADO ACTIVIDAD CERCA DE TU VEHICULO!");
//text content
    updateSerial();
    mySerial.write(26);
    Serial.println("Message sent!");
    delay(1000);
}
}
digitalWrite(LedRed, LOW);
digitalWrite(LedGreen, HIGH);
digitalWrite(LedYellow, LOW);
delay(25);
}
//Primer sensor ultrasónico
int CalcularDistancia_1(){
    digitalWrite(TRIGGER1, LOW);
    delayMicroseconds(4);
    digitalWrite(TRIGGER1, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIGGER1, LOW);

    tiempo1 = pulseIn(ECH01, HIGH);
    distancia1 = tiempo1 / 58.2;
    return distancia1;
```



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
}  
//Segundo sensor ultrasónico  
int CalcularDistancia_2(){  
    digitalWrite(TRIGGER2, LOW);  
    delayMicroseconds(4);  
    digitalWrite(TRIGGER2, HIGH);  
    delayMicroseconds(10);  
    digitalWrite(TRIGGER2, LOW);  
  
    tiempo2 = pulseIn(ECHO2, HIGH);  
    distancia2 = tiempo2 / 58.2;  
    return distancia2;  
}  
//tercer sensor ultrasónico  
int CalcularDistancia_3(){  
    digitalWrite(TRIGGER3, LOW);  
    delayMicroseconds(4);  
    digitalWrite(TRIGGER3, HIGH);  
    delayMicroseconds(10);  
    digitalWrite(TRIGGER3, LOW);  
  
    tiempo3 = pulseIn(ECHO3, HIGH);  
    distancia3 = tiempo3 / 58.2;  
    return distancia3;  
}  
  
//Funciones para las interrupciones  
  
//Tiempo deseado para mandar el mensaje  
int buttonPressed(){  
if(millis() - timePassed >= timeInterval){  
    timePassed = millis();  
    tiempoMSG += 5;  
    if(tiempoMSG == 25)  
        tiempoMSG = 10;  
    if(tiempoMSG == 10){  
        analogWrite(LedRed_RGB, 255);  
        analogWrite(LedGreen_RGB, 255);  
        analogWrite(LedBlue_RGB, 255);  
    }  
    else if(tiempoMSG == 15){  
        analogWrite(LedRed_RGB, 0);  
        analogWrite(LedGreen_RGB, 255);  
        analogWrite(LedBlue_RGB, 255);  
    }  
    else if(tiempoMSG == 20){  
        analogWrite(LedRed_RGB, 255);  
        analogWrite(LedGreen_RGB, 0);  
        analogWrite(LedBlue_RGB, 0);  
    }  
}
```



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
}  
}  
return tiempoMSG;  
}  
  
void updateSerial()  
{  
    delay(1000);  
    while (Serial.available())  
    {  
        mySerial.write(Serial.read()); //Forward what Serial received to Software Serial  
Port  
    }  
    while(mySerial.available())  
    {  
        Serial.write(mySerial.read()); //Forward what Software Serial received to Serial  
Port  
    }  
}
```

11.2. Código de Android Studio:

Activity Splash:

```
package com.example.seguridad
```

```
import android.content.Intent  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.os.Handler  
import android.os.PersistableBundle  
import android.view.animation.AnimationUtils  
import kotlinx.android.synthetic.main.activity_splash_screen.*
```

```
class SplashScreen : AppCompatActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_splash_screen)  
  
        val topAnimation = AnimationUtils.loadAnimation(this, R.anim.top_animation)  
        val middleAnimation = AnimationUtils.loadAnimation(this,  
R.anim.top_animation)  
        val bottonAnimation = AnimationUtils.loadAnimation(this,  
R.anim.botton_animation)  
  
        safety.startAnimation(topAnimation)  
        imagen3.startAnimation(middleAnimation)  
        t7.startAnimation(bottonAnimation)
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
val splashScreenTimeOut = 4000
val homeIntent:Intent = Intent(this@SplashScreen, MainActivity::class.java)

Handler().postDelayed({
    startActivity(homeIntent)
    finish()
}, splashScreenTimeOut.toLong())
}
}
```

Activity Bienvenida:

```
package com.example.seguridad
```

```
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.EditText
import kotlinx.android.synthetic.main.activity_main.*
```

```
class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        supportActionBar?.hide()

        B1.setOnClickListener{
            val intent:Intent = Intent(this, Registro::class.java)
            startActivity(intent)
        }

        b2.setOnClickListener{
            val intent:Intent = Intent(this, Datos::class.java)
            startActivity(intent)
        }
    }
}
```

Activity Registro:

```
package com.example.seguridad
```

```
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.Handler
import android.view.View
import android.view.animation.AnimationUtils
import androidx.appcompat.app.AlertDialog
import com.google.android.gms.auth.api.signin.GoogleSignIn
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
import com.google.android.gms.auth.api.signin.GoogleSignInOptions
import com.google.android.gms.common.api.ApiException
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.auth.GoogleAuthProvider
import kotlinx.android.synthetic.main.activity_datos.*
import kotlinx.android.synthetic.main.activity_datos.name_b
import kotlinx.android.synthetic.main.activity_datos.titulo
import kotlinx.android.synthetic.main.alert1.view.*
import kotlinx.android.synthetic.main.alert2.view.*
import kotlinx.android.synthetic.main.alert4.view.*
import kotlinx.android.synthetic.main.alert5.view.*
import kotlinx.android.synthetic.main.alert7.view.*
import kotlinx.android.synthetic.main.alert8.view.*
import java.util.regex.Pattern

class Datos : AppCompatActivity() {
    private val GOOGLE_SIGN = 100

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_datos)
        supportActionBar?.hide()

        //Proceso de registro de usuario
        setup()

        val topAnimation = AnimationUtils.loadAnimation(this, R.anim.top2_animation)

        imagen.startAnimation(topAnimation)
        titulo.startAnimation(topAnimation)
        layoutemail.startAnimation(topAnimation)
        layoutcon.startAnimation(topAnimation)
        name_b.startAnimation(topAnimation)
        entrar.startAnimation(topAnimation)
        textU.startAnimation(topAnimation)
        google.startAnimation(topAnimation)
        layoutcon2.startAnimation(topAnimation)
    }

    private fun setup(){
        val passwordRegex = Pattern.compile(
            "^" + "(?=.*[0-9])" +
                "(?=.*[a-z])" +
                "(?=.*[A-Z])" +
                "(?=.*[!@#$%^&+=-])" +
                "(?=.*\\S+$)" +
                ".{4,}" +
                "$"
        )
    }
}
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
entrar.setOnClickListener{
    if (email.text.isNotEmpty() && c1.text.isNotEmpty() &&
c2.text.isNotEmpty()){
        if (c1.text.toString() == c2.text.toString()) {
            FirebaseAuth.getInstance()
                .createUserWithEmailAndPassword(
                    email.text.toString(),
                    c1.text.toString()
                ).addOnCompleteListener {
                    if (it.isSuccessful) {
                        val loading = LoadingDialog(this)
                        loading.startLoading()
                        val handler = Handler()
                        handler.postDelayed(object : Runnable{
                            override fun run(){
                                loading.isDismiss()
                                home(it.result?.user?.email ?: "",
ProviderType.BASIC)
                            }
                        }, 5000)
                    } else {
                        alert()
                    }
                }
            } else {
                val view = View.inflate(this, R.layout.alert7, null)
                val builder = AlertDialog.Builder(this)
                builder.setView(view)
                val dialog: AlertDialog = builder.create()
                dialog.show()

                dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
                view.btnOk7.setOnClickListener {
                    dialog.dismiss()
                }
            }
        } else{
            val view = View.inflate(this, R.layout.alert2, null)
            val builder = AlertDialog.Builder(this)
            builder.setView(view)
            val dialog: AlertDialog = builder.create()
            dialog.show()

            dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
            view.btnOk2.setOnClickListener {
                dialog.dismiss()
            }
        }
    }
}
```




INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
google.setOnClickListener{
    val ini =
GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT_SIGN_IN).requestIdToken(getSt
ring(R.string.default_web_client_id))
    .requestEmail()
    .build()

    val cliente = GoogleSignIn.getClient(this, ini)
    cliente.signOut()
    startActivityForResult(cliente.signInIntent, GOOGLE_SIGN)
}
}

private fun alert(){
    val view = View.inflate(this, R.layout.alert4, null)
    val builder = AlertDialog.Builder(this)
    builder.setView(view)
    val dialog: AlertDialog = builder.create()
    dialog.show()
    dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnOk4.setOnClickListener {
        dialog.dismiss()
    }
}

private fun home(email: String, provider: ProviderType){
    val intent: Intent = Intent(this, Menu::class.java).apply {
    }
    intent.putExtra("email2", email)
    startActivity(intent)
    finish()
}

override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
    super.onActivityResult(requestCode, resultCode, data)
    if(requestCode == GOOGLE_SIGN){
        val task = GoogleSignIn.getSignedInAccountFromIntent(data)
        try{
            val cuenta = task.getResult(ApiException::class.java)
            if(cuenta != null) {
                val credencial =
GoogleAuthProvider.getCredential(cuenta.idToken, null)

                FirebaseAuth.getInstance().signInWithCredential(credencial).addOnCompleteListener {
                    if(it.isSuccessful){
                        val loading = LoadingDialog(this)
                        loading.startLoading()
                        val handler = Handler()
                        handler.postDelayed(object :Runnable{
```



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
        override fun run(){
            loading.isDismiss()
            home(it.result?.user?.email ?: "",
ProviderType.BASIC)
        }
    }, 5000)
} else {
    alert()
}
}
}
} catch (e: ApiException){
    val view = View.inflate(this, R.layout.alert5, null)
    val builder = AlertDialog.Builder(this)
    builder.setView(view)
    val dialog: AlertDialog = builder.create()
    dialog.show()

    dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnOk5.setOnClickListener {
        dialog.dismiss()
    }
}
}
}
```

Activity Iniciar Sesión:

```
package com.example.seguridad
```

```
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.Handler
import android.view.LayoutInflater
import android.view.View
import android.view.animation.AnimationUtils
import androidx.appcompat.app.AlertDialog
import com.google.android.gms.auth.api.signin.GoogleSignIn
import com.google.android.gms.auth.api.signin.GoogleSignInOptions
import com.google.android.gms.common.api.ApiException
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.auth.GoogleAuthProvider
import kotlinx.android.synthetic.main.activity_registro.*
import kotlinx.android.synthetic.main.activity_registro.name_b
import kotlinx.android.synthetic.main.alert1.*
import kotlinx.android.synthetic.main.alert2.view.*
import kotlinx.android.synthetic.main.alert3.view.*
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
class Registro : AppCompatActivity() {
    private val GOOGLE_SIGN = 100
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_registro)
        supportActionBar?.hide()
        //setup
        setup()

        name_b.setOnClickListener {
            val intent:Intent = Intent(this, Contra::class.java)
            startActivity(intent)
        }

        val topAnimation = AnimationUtils.loadAnimation(this, R.anim.top2_animation)

        imagen1.startAnimation(topAnimation)
        titulo2.startAnimation(topAnimation)
        layoutema.startAnimation(topAnimation)
        layoutemail.startAnimation(topAnimation)
        name_b.startAnimation(topAnimation)
        btn_ini.startAnimation(topAnimation)
        t2.startAnimation(topAnimation)
        google1.startAnimation(topAnimation)
    }

    private fun setup(){
        btn_ini.setOnClickListener {
            if (email.text.isNotEmpty() && contra.text.isNotEmpty()){
                FirebaseAuth.getInstance().signInWithEmailAndPassword(email.text.toString(),
                contra.text.toString()).addOnCompleteListener{
                    if (it.isSuccessful){
                        val loading = LoandigDialog(this)
                        loading.startLoading()
                        val handler = Handler()
                        handler.postDelayed(object :Runnable{
                            override fun run(){
                                loading.isDismiss()
                                home()
                            }
                        }, 5000)
                    }else{
                        alert()
                    }
                }
            }else{
                val view = View.inflate(this, R.layout.alert2, null)
            }
        }
    }
}
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
val builder = AlertDialog.Builder(this)
builder.setView(view)
val dialog: AlertDialog = builder.create()
dialog.show()
```

```
dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnOk2.setOnClickListener {
        dialog.dismiss()
    }
}

google1.setOnClickListener{
    val ini =
    GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT_SIGN_IN).requestIdToken(getSt
ring(R.string.default_web_client_id))
        .requestEmail()
        .build()

    val cliente = GoogleSignIn.getClient(this, ini)
    cliente.signOut()
    startActivityForResult(cliente.signInIntent, GOOGLE_SIGN)
}
}
private fun alert(){
    val view = View.inflate(this, R.layout.alert1, null)
    val builder = AlertDialog.Builder(this)
    builder.setView(view)
    val dialog: AlertDialog = builder.create()
    dialog.show()
    dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.bOk.setOnClickListener {
        dialog.dismiss()
    }
}
private fun home(){
    val intent:Intent = Intent(this, Menu::class.java)
    intent.putExtra("email1", email.text.toString())
    startActivity(intent)
    finish()
}
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
    super.onActivityResult(requestCode, resultCode, data)
    if(requestCode == GOOGLE_SIGN){
        val task = GoogleSignIn.getSignedInAccountFromIntent(data)
        try{
            val cuenta = task.getResult(ApiException::class.java)
            if(cuenta != null) {
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
    val credencial = FirebaseAuthProvider.getCredential(cuenta.idToken,
null)

FirebaseAuth.getInstance().signInWithCredential(credencial).addOnCompleteListener {
    if(it.isSuccessful){
        val loading = LoadingDialog(this)
        loading.startLoading()
        val handler = Handler()
        handler.postDelayed(object :Runnable{
            override fun run(){
                loading.dismiss()
                home()
            }
        }, 5000)
    }else{
        alert()
    }
}
}
}catch (e: ApiException){
    val view = View.inflate(this, R.layout.alert3, null)
    val builder = AlertDialog.Builder(this)
    builder.setView(view)
    val dialog: AlertDialog = builder.create()
    dialog.show()

    dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnOk3.setOnClickListener {
        dialog.dismiss()
    }
}
}
}
}
```

Activity recuperar contraseña:

```
package com.example.seguridad
```

```
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.view.animation.AnimationUtils
import android.widget.Toast
import androidx.appcompat.app.AlertDialog
import com.google.firebase.auth.FirebaseAuth
import kotlinx.android.synthetic.main.activity_contra.*
import kotlinx.android.synthetic.main.alert2.view.*
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
import kotlinx.android.synthetic.main.alert6.view.*

class Contra : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_contra)
        supportActionBar?.hide()

        Restaurar()

        val topAnimation = AnimationUtils.loadAnimation(this, R.anim.top2_animation)
        titulo.startAnimation(topAnimation)
        email2.startAnimation(topAnimation)
        imagen.startAnimation(topAnimation)
        btn_crear.startAnimation(topAnimation)
        textU.startAnimation(topAnimation)
    }

    private fun Restaurar(){
        btn_crear.setOnClickListener{
            if (email2.text.toString().isEmpty()){
                val view = View.inflate(this, R.layout.alert2, null)
                val builder = AlertDialog.Builder(this)
                builder.setView(view)
                val dialog: AlertDialog = builder.create()
                dialog.show()

                dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
                view.btnOk2.setOnClickListener {
                    dialog.dismiss()
                }
            }
            if (email2.text.toString().isNotEmpty()){
                FirebaseAuth.getInstance().sendPasswordResetEmail(email2.text.toString()).addOnCompleteListener{
                    if (it.isSuccessful){
                        Toast.makeText(this, "Email Enviado", Toast.LENGTH_LONG).show()
                        val intent:Intent = Intent(this, Registro::class.java)
                        startActivity(intent)
                    }
                    else{
                        val view = View.inflate(this, R.layout.alert6, null)
                        val builder = AlertDialog.Builder(this)
                        builder.setView(view)
                        val dialog: AlertDialog = builder.create()
                        dialog.show()

                        dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
                    }
                }
            }
        }
    }
}
```



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
view.btnOk6.setOnClickListener {  
    dialog.dismiss()  
}  
}  
}  
}  
}  
}
```

Activity Menu:

```
package com.example.seguridad
```

```
import android.content.Intent  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.View  
import android.view.animation.AnimationUtils  
import androidx.appcompat.app.AlertDialog  
import com.google.firebase.auth.FirebaseAuth  
import kotlinx.android.synthetic.main.activity_menu.*  
import kotlinx.android.synthetic.main.loginout.view.*
```

```
enum class ProviderType{  
    BASIC,  
    GOOGLE  
}
```

```
class Menu : AppCompatActivity() {  
    private val store = FirebaseAuth.getInstance()  
  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_menu)  
        supportActionBar?.hide()  
  
        val bundle = intent.extras  
        val dato:String? = bundle?.getString("email2")  
        val dato2:String? = bundle?.getString("email1")  
        val n1:String? = bundle?.getString("nombre1")  
  
        b1.setOnClickListener(){  
            val intent:Intent = Intent(this, datos2::class.java)  
            startActivity(intent)  
        }  
  
        b4.setOnClickListener(){  
            val intent:Intent = Intent(this, cambio::class.java)
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
startActivity(intent)
}

b6.setOnClickListener{
    val view = View.inflate(this, R.layout.Loginout, null)
    val builder = AlertDialog.Builder(this)
    builder.setView(view)
    val dialog: AlertDialog = builder.create()
    dialog.show()
    dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnSi.setOnClickListener {
        FirebaseAuth.getInstance().signOut()
        val intent:Intent = Intent(this, MainActivity::class.java)
        startActivity(intent)
        finish()
    }
    view.btnNo.setOnClickListener {
        dialog.dismiss()
    }
}
b2.setOnClickListener{
    val intent:Intent = Intent(this, galeria::class.java)
    startActivity(intent)
}
b7.setOnClickListener{
    val intent:Intent = Intent(this, Video::class.java)
    startActivity(intent)
}

val topAnimation = AnimationUtils.loadAnimation(this, R.anim.top2_animation)

b1.startAnimation(topAnimation)
b2.startAnimation(topAnimation)
b4.startAnimation(topAnimation)
b6.startAnimation(topAnimation)
b7.startAnimation(topAnimation)
textM.startAnimation(topAnimation)
textVi2.startAnimation(topAnimation)
Datos_1.startAnimation(topAnimation)
contra_1.startAnimation(topAnimation)
textS.startAnimation(topAnimation)
Titulo_Safety.startAnimation(topAnimation)
}
}
```

Activity Datos del usuario:



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
package com.example.seguridad
```

```
import android.app.Activity
import android.content.Intent
import android.net.Uri
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.provider.MediaStore
import android.view.View
import android.widget.AdapterView
import android.widget.AdapterView.OnItemClickListener
import android.widget.ArrayAdapter
import android.widget.Spinner
import androidx.appcompat.app.AlertDialog
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FirebaseFirestore
import kotlinx.android.synthetic.main.activity_datos2.*
import kotlinx.android.synthetic.main.activity_datos2.apellidos
import kotlinx.android.synthetic.main.activity_datos2.fecha1
import kotlinx.android.synthetic.main.activity_datos2.name
import kotlinx.android.synthetic.main.activity_datos2.telefono
import kotlinx.android.synthetic.main.alert2.view.*

class datos2 : AppCompatActivity() {
    private val store = FirebaseAuth.getInstance()
    private val db = FirebaseFirestore.getInstance()
    private val userid = store.currentUser?.uid
    lateinit var filepath : Uri

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_datos2)
        this.setTitle(R.string.datos_1)

        val spinner = findViewById<Spinner>(R.id.be)
        val lista = resources.getStringArray(R.array.Estados)
        val adaptador = ArrayAdapter(this, R.layout.spinner, lista)
        spinner.adapter = adaptador
        spinner.onItemSelectedListener = object : AdapterView.OnItemSelectedListener{
            override fun onNothingSelected(p0: AdapterView<*>?) {
            }
            override fun onItemSelected(p0: AdapterView<*>?, p1: View?, p2: Int, p3:
Long) {
                estado.text = lista.get(p2)
            }
        }

        val spinner2 = findViewById<Spinner>(R.id.bg)
        val lista2 = resources.getStringArray(R.array.gener)
        val adaptador2 = ArrayAdapter(this, R.layout.spinner, lista2)
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
spinner2.adapter = adaptador2
```

```
spinner2.onItemSelectedListener = object :  
AdapterView.OnItemSelectedListener{  
    override fun onNothingSelected(p0: AdapterView<*>?) {  
    }  
    override fun onItemSelected(p0: AdapterView<*>?, p1: View?, p2: Int, p3:  
Long) {  
        genero.text = lista2.get(p2)  
    }  
}  
  
fecha1.setOnClickListener{showDatePickerDialog()}  
  
db.collection("Users").document(userid).get().addOnSuccessListener {  
    name.setText(it.get("name") as String?)  
    apellidos.setText(it.get("apellidos") as String?)  
    telefono.setText(it.get("telefono") as String?)  
    fecha1.setText(it.get("fecha") as String?)  
    ciudad.setText(it.get("ciudad") as String?)  
    genero.setText(it.get("genero") as String?)  
    estado.setText(it.get("estado") as String?)  
}  
  
val bundle = intent.extras  
val dato:String? = bundle?.getString("email1")  
val dato2:String? = bundle?.getString("email2")  
  
be.isEnabled = false  
bg.isEnabled = false  
boton_Ima.isEnabled = false  
  
cambiar.setOnClickListener{  
    name.isEnabled = true  
    apellidos.isEnabled = true  
    telefono.isEnabled = true  
    fecha1.isEnabled = true  
    bg.isEnabled = true  
    be.isEnabled = true  
    ciudad.isEnabled = true  
    genero.isEnabled = true  
    boton_Ima.isEnabled =true  
}  
  
guardar.setOnClickListener {  
    if (name.text.isNotEmpty() && apellidos.text.isNotEmpty() &&  
telefono.text.isNotEmpty() && fecha1.text.isNotEmpty() && ciudad.text.isNotEmpty())
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
&& genero.text.isNotEmpty()) {
    val intent: Intent = Intent(this, Menu::class.java)
    intent.putExtra("nombre1", name.text.toString())
    startActivity(intent)
    name.isEnabled = false
    apellidos.isEnabled = false
    telefono.isEnabled = false
    fecha1.isEnabled = false
    bg.isEnabled = false
    be.isEnabled = false
    ciudad.isEnabled = false
    genero.isEnabled = false
    estado.isEnabled = false
    boton_Ima.isEnabled = false

    db.collection("Users").document(userid).set(
        hashMapOf(
            "name" to name.text.toString(),
            "apellidos" to apellidos.text.toString(),
            "telefono" to telefono.text.toString(),
            "fecha" to fecha1.text.toString(),
            "genero" to genero.text.toString(),
            "estado" to estado.text.toString(),
            "ciudad" to ciudad.text.toString()
        )
    )

    //subir_image()
} else {
    val view = View.inflate(this, R.layout.alert2, null)
    val builder = AlertDialog.Builder(this)
    builder.setView(view)
    val dialog: AlertDialog = builder.create()
    dialog.show()

    dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnOk2.setOnClickListener {
        dialog.dismiss()
    }
}

}

}

private fun showDatePickerDialog() {
    val datePicker = DatePickerFragment{day, month, year ->onDateSelected(day,
month, year)}
    datePicker.show(supportFragmentManager, "datePicker")
}

fun onDateSelected(day: Int, month: Int, year: Int){
```



PROYECTO INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
fecha1.setText(" "+ day + "/" + (month+1) + "/" + year)
}

private fun cam_image(){
    var i = Intent()
    i.setType("Image/*")
    i.setAction(Intent.ACTION_GET_CONTENT)
    startActivityForResult(Intent.createChooser(i, "cambio de imagen"), 111)
}

override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
    super.onActivityResult(requestCode, resultCode, data)
    if (requestCode==111 && resultCode== Activity.RESULT_OK && data != null){
        filepath = data.data!!
        var bitmap = MediaStore.Images.Media.getBitmap(contentResolver, filepath)
        boton_Ima.setImageBitmap(bitmap)
    }
}
}
```

Activity Video:

```
package com.example.seguridad

import android.app.DownloadManager
import android.content.Context
import android.net.Uri
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.Environment
import android.widget.Toast
import com.google.firebase.storage.FirebaseStorage
import com.google.firebase.storage.StorageReference
import kotlinx.android.synthetic.main.activity_video.*
import java.io.File
import java.lang.Exception

class Video : AppCompatActivity() {
    private var fileRef : StorageReference? = null
    var mydownloadid:Long = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_video)
        this.setTitle("Videos")

        downloadVideo()

        download.setOnClickListener {
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
        saveVideo()
    }
}
private fun downloadVideo() {
    try {
        val storage = FirebaseStorage.getInstance().getReference()
        val videoRef = storage.child("video_vigilancia.mp4")
        val localFile = File.createTempFile("video_vigilancia", "mp4")

        videoRef.getFile(localFile).addOnSuccessListener {
            videoView.setVideoURI(Uri.fromFile(localFile))
            videoView.start()
        }
    } catch (e: Exception){
        Toast.makeText(this, "Fallo" + e.localizedMessage,
Toast.LENGTH_LONG).show()
    }
}

private fun saveVideo(){
    var request =
DownloadManager.Request(Uri.parse("https://firebasestorage.googleapis.com/v0/b/security-car-e36e9.appspot.com/o/video_vigilancia.mp4?alt=media&token=a4e5176d-d8b8-4e0a-b667-2ec8a350b3ff"))
        .setTitle("Video-Seguridad")
        .setDescription("Descargando...")

.setNotificationVisibility(DownloadManager.Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETE
D)
        .setAllowedOverMetered(true)
        .setDestinationInExternalPublicDir(Environment.DIRECTORY_DOWNLOADS,
"${System.currentTimeMillis()}")

    var dm = getSystemService(Context.DOWNLOAD_SERVICE) as DownloadManager
    mydownloadid = dm.enqueue(request)
}
}
```

Activity Imagen:

```
package com.example.seguridad
```

```
import android.app.DownloadManager
import android.content.*
import android.graphics.Bitmap
import android.graphics.BitmapFactory
import android.net.Uri
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.Environment
```



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
import android.view.animation.RotateAnimation
import android.widget.Toast
import com.google.firebase.ktx.Firebase
import com.google.firebase.storage.FirebaseStorage
import com.google.firebase.storage.StorageReference
import com.google.firebase.storage.ktx.storage
import kotlinx.android.synthetic.main.activity_galeria.*

class galeria : AppCompatActivity() {
    private var fileUri: Uri? = null
    private var bitmap: Bitmap? = null
    private var imageReference: StorageReference? = null
    private var fileRef : StorageReference? = null
    var mydownloadid:Long = 0
    val image = Firebase.storage.reference

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_galeria)
        this.setTitle("Galeria")

        loadImage()

        download2.setOnClickListener {
            saveImage()
        }

        var br = object:BroadcastReceiver(){
            override fun onReceive(p0: Context?, p1: Intent?) {
                var id = p1?.getLongExtra(DownloadManager.EXTRA_DOWNLOAD_ID, -1)
                if (id ==mydownloadid){
                    Toast.makeText(applicationContext, "Descarga Completa",
                    Toast.LENGTH_SHORT).show()
                }
            }
        }
        registerReceiver(br, IntentFilter(DownloadManager.ACTION_DOWNLOAD_COMPLETE))
    }

    private fun loadImage() {
        val storage = FirebaseStorage.getInstance()
        val imageRef = storage.getReference().child("movement.jpg")

        imageRef.getBytes(1024*1024).addOnSuccessListener { bytes ->
            val bitmap = BitmapFactory.decodeByteArray(bytes, 0, bytes.size)
            imageref.setImageBitmap(bitmap)
        }
    }
}
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
private fun saveImage(){
    var request =
DownloadManager.Request(Uri.parse("https://firebasestorage.googleapis.com/v0/b/security-car-e36e9.appspot.com/o/movement.jpg?alt=media&token=d7cf7d30-ca42-4d15-bb8b-49d33743e785"))
        .setTitle("Imagen-Seguridad")
        .setDescription("Descargando...")

.setNotificationVisibility(DownloadManager.Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETED)

        .setAllowedOverMetered(true)
        .setDestinationInExternalPublicDir(Environment.DIRECTORY_DOWNLOADS,
"${System.currentTimeMillis()}")

        var dm = getSystemService(Context.DOWNLOAD_SERVICE) as DownloadManager
        mydownloadid = dm.enqueue(request)
    }
}
```

Activity cambio de contraseña:

```
package com.example.seguridad
```

```
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.Toast
import androidx.appcompat.app.AlertDialog
import com.google.firebase.auth.FirebaseAuth
import kotlinx.android.synthetic.main.activity_cambio.*
import kotlinx.android.synthetic.main.activity_contra.btn_crear
import kotlinx.android.synthetic.main.alert2.view.*
import kotlinx.android.synthetic.main.alert6.view.*
```

```
class cambio : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_cambio)
        this.setTitle(R.string.contraseña_cam)

        Cambiar()
    }

    private fun Cambiar(){
        btn_crear.setOnClickListener{
            if (email.text.toString().isEmpty()){
                val view = View.inflate(this, R.layout.alert2, null)
                val builder = AlertDialog.Builder(this)
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
builder.setView(view)
val dialog: AlertDialog = builder.create()
dialog.show()
```

```
dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnOk2.setOnClickListener {
        dialog.dismiss()
    }
}
if (email.text.toString().isEmpty()){
```

```
FirebaseAuth.getInstance().sendPasswordResetEmail(email.text.toString()).addOnCompleteListener{
```

```
    if (it.isSuccessful){
        Toast.makeText(this, "Email Enviado",
Toast.LENGTH_LONG).show()
        val intent:Intent = Intent(this, Registro::class.java)
        startActivity(intent)
    }
    else{
        val view = View.inflate(this, R.layout.alert6, null)
        val builder = AlertDialog.Builder(this)
        builder.setView(view)
        val dialog: AlertDialog = builder.create()
        dialog.show()
    }
}
```

```
dialog.window?.setBackgroundDrawableResource(android.R.color.transparent)
    view.btnOk6.setOnClickListener {
        dialog.dismiss()
    }
}
}
}
}
}
}
```

código para mostrar calendario:

```
package com.example.seguridad
```

```
import android.app.DatePickerDialog
import android.app.Dialog
import android.content.Context
import android.os.Bundle
import android.widget.DatePicker
import androidx.fragment.app.DialogFragment
import java.util.*
```




INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
class DatePickerFragment (val listener: (day:Int, month:Int, year:Int) -> Unit):  
DialogFragment(),  
    DatePickerDialog.OnDateSetListener {  
    override fun onDateSet(view: DatePicker?, year: Int, month: Int, dayOfMonth: Int)  
{  
        listener(dayOfMonth, month, year)  
    }  
  
    override fun onCreateDialog(savedInstanceState: Bundle?): Dialog {  
        val c = Calendar.getInstance()  
        val day = c.get(Calendar.DAY_OF_MONTH)  
        val month = c.get(Calendar.MONTH)  
        val year = c.get(Calendar.YEAR)  
  
        val picker = DatePickerDialog(activity as Context,  
R.style.calendarColor, this, year, month, day)  
        picker.datePicker.maxDate = c.timeInMillis  
        return picker  
    }  
}
```

Código para mostrar mensaje, cargando perfil:

```
package com.example.seguridad
```

```
import android.app.Activity  
import android.app.AlertDialog
```

```
class LoandigDialog(val mActivity:Activity) {  
    private lateinit var isdialog:AlertDialog  
    fun startLoanding(){  
        val infalter = mActivity.layoutInflater  
        val dialogView = infalter.inflate(R.layout.barra, null)  
        val builder = AlertDialog.Builder(mActivity)  
        builder.setView(dialogView)  
        builder.setCancelable(false)  
        isdialog = builder.create()  
        isdialog.show()  
        isdialog.window?.setBackgroundDrawableResource(android.R.color.transparent)  
    }  
    fun isDismiss(){  
        isdialog.dismiss()  
    }  
}
```

11.3. Código Raspberry pi 4:

```
from picamera import PiCamera  
from time import sleep  
import smtplib
```



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
gmail_user = 'envio.asistente.video.vigilancia@gmail.com'
gmail_password = 'aprobado21'
import time
from datetime import datetime
from email.mime.image import MIMEImage
from email.mime.multipart import MIMEMultipart
import RPi.GPIO as GPIO
import time

from subprocess import call
import os
import email.mime.application
import datetime
import smtplib
import serial
import pyrebase

toaddr = 'asistente.video.vigilancia@gmail.com'
me = 'envio.asistente.video.vigilancia@gmail.com'
Subject='security alert'

camera = PiCamera()

GPIO.setmode(GPIO.BCM)

#P=PiCamera()
#P.resolution= (1024,768)
#P.start_preview()
firebaseConfig = {
    'apiKey': "AIzaSyCcoW3SnGbbjDgdxY0AeF9qpnhVMu5BrSU",
    'authDomain': "security-car-e36e9.firebaseio.com",
    'databaseURL': "https://security-car-e36e9.appspot.com",
    'projectId': "security-car-e36e9",
    'storageBucket': "security-car-e36e9.appspot.com",
    'messagingSenderId': "1003061248625",
    'appId': "1:1003061248625:web:0f22f48c9e1cb43dcb51d0",
    'measurementId': "G-P3MM4448CN"
}
firebase = pyrebase.initialize_app(firebaseConfig)
storage = firebase.storage()

GPIO.setup(23, GPIO.IN)
while True:
    if GPIO.input(23):
        print("Movimiento detectado")
        #Video record
        camera.resolution = (640,480)
        camera.rotation = 180
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

```
camera.start_recording('video_vigilancia.h264')
camera.wait_recording(30)
camera.stop_recording()

#converting video from .h264 to .mp4
command = "MP4Box -add video_vigilancia.h264 video_vigilancia.mp4"
call([command], shell=True)
print("video convertido")
storage.child('video_vigilancia.mp4').put('video_vigilancia.mp4')
print("Video sent")

#Create the Message
subject='Security allert!!'
msg = MIMEMultipart()
msg['Subject'] = subject
msg['From'] = me
msg['To'] = toaddr
    #Video
Captured = '/home/pi/Desktop/video_vigilancia.mp4'
fp=open(Captured,'rb')
att = email.mime.application.MIMEApplication(fp.read(),_subtype=".mp4")
fp.close()
att.add_header('Content-Disposition','attachment',filename='video' +
datetime.datetime.now().strftime('%Y-%m-%d%H:%M:%S') + '.mp4')
msg.attach(att)
print("Grabacion exitosa")
print("Motion...")
#camera warm-up time
time.sleep(1)
camera.capture('movement.jpg')
storage.child('movement.jpg').put('movement.jpg')
print("Image sent")
time.sleep(4)
subject='Security allert!!'
msg = MIMEMultipart()
msg['Subject'] = subject
msg['From'] = me
msg['To'] = toaddr

fp= open('movement.jpg','rb')
img = MIMEImage(fp.read())
fp.close()
msg.attach(img)

#removing .h264 & .mp4 extra files
os.remove("/home/pi/Desktop/video_vigilancia.h264")
#renaming file
```



INGENIERÍA EN SISTEMAS AUTOMOTRICES

Referencias

- [1] DIEGO AGUILAR Y BELÉN SALDÍVAR. (2019). ROBO DE VEHÍCULOS ASEGURADOS. 05/02/2021, DE ELECONOMISTA SITIO WEB: <HTTPS://WWW.ELECONOMISTA.COM.MX/EMPRESAS/ROBO-DE-VEHICULOS-ASEGURADOS-DISMINUYO-11-ANUAL-EN-EL-2019-AMIS-20200206-0078.HTML>
- [2] MARIANA VERGARA. (2015). 10 PERCANCES QUE LOS SEGUROS DE TUS AUTOS NO CUMPLEN. 05/02/2021, DE ELECONOMISTA SITIO WEB: <HTTPS://WWW.ELECONOMISTA.COM.MX/FINANZASPERSONALES/10-PERCANCES-QUE-LOS-SEGUROS-DE-AUTO-NO-CUBREN-20150509-0012.HTML>
- [3] DÉBORA BRIZUELA. (2019). ¿ME CUBRE EL SEGURO SI ME RAYAN EL AUTO?. 05/02/2021, DE COMPRAENCASA SITIO WEB: <HTTPS://WWW.COMPARAENCASA.COM/SEGUROS-DE-AUTO/SEGUROS-DEL-AUTOMOTOR/ME-CUBRE-EL-SEGURO-SI-ME-RAYAN-EL-AUTO/#:~:TEXT=GENERALMENTE%2C%20LAS%20ASEGURADORAS%20NO%20CUBRE%20N,A%20PROP%C3%B3SITO%20POR%20TOTAL%20NEGLIGENCIA>
- [4] GERTRUDIS GOMEZ. (15 DE ENERO DE 2018). CARCENTINEL. 15 DE ENERO DE 2018, DE CARCENTINEL SITIO WEB: <HTTPS://PROENIUM.COM/CARCENTINEL/>
- [5] LUCAS NUNES DA SILVA NETO (2016). ASISTENTE DE SEGURIDAD PARA UN AUTOMÓVIL (TESIS DE PREGRADO, MÁSTER EN INGENIERÍA DE COMPUTADORES Y REDES). UNIVERSIDAD POLITÉCNICA DE VALENCIA, VALENCIA, ESPAÑA
- [6] SALDAÑA PÉREZ ANA MARÍA MAGDALENA & XOCHIHUA JORGE ALEJANDRO (2011). ASISTENTE CONTROL DE ACCESO A ESTACIONAMIENTO POR MEDIO DEL RECONOCIMIENTO DE PLACAS VEHICULARES (TESIS DE LICENCIATURA, INGENIERO EN COMUNICACIONES Y ELECTRÓNICA). INSTITUTO POLITÉCNICO NACIONAL, MÉXICO
- [7] ALAN TOFTE. (23 DE AGOSTO DE 2016). ADAPTIVE RECOGNITION. 23 DE AGOSTO DE 2016, DE ADAPTIVE RECOGNITION HUNGARY SITIO WEB: <HTTPS://ADAPTIVERECOGNITION.COM/>
- [8] JESÚS HÉCTOR MEDINA CERVANTES (2009). ASISTENTE DE SEGURIDAD



INGENIERÍA EN SISTEMAS AUTOMOTRICES

AUTOMOTRIZ 3G/GSM (TESIS DE LICENCIATURA, INGENIERO INDUSTRIAL).
UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO, MÉXICO.

[9] DARWIN S. ALDÁS S., SANTIAGO M. COLLANTES V. Y JOHN P. REYES V. (2016). PROCESAMIENTO DE IMÁGENES CON VISIÓN ARTIFICIAL PARA EL RECONOCIMIENTO DE PLACAS VEHICULARES (TESIS DE LICENCIATURA, INGENIERO MECATRONICO). UNIVERSIDAD TÉCNICA DE AMBATO, ECUADOR.

[10] MUNDACA, G. (2016). DETECCIÓN DE CARACTERES DE PLACAS DE AUTOMÓVILES MEDIANTE TÉCNICAS DE VISIÓN ARTIFICIAL (TESIS DE PREGRADO EN INGENIERÍA MECÁNICO-ELÉCTRICA). UNIVERSIDAD DE PIURA. FACULTAD DE INGENIERÍA. PROGRAMA ACADÉMICO DE INGENIERÍA MECÁNICO-ELÉCTRICA. PIURA, PERÚ.

[11] MILTON DANIEL TORRES CARRERA (2020). RECONOCIMIENTO AUTOMÁTICO DE LA PLACA DE UN VEHÍCULO DE ECUADOR (TESIS PARA OBTENER EL TÍTULO DE INGENIERO EN ASISTENTES). UNIVERSIDAD POLITECNICA SALESIANA. QUITO, ECUADOR.

[12] TESLA. (2021). AUTOPILOT. 2021, DE TESLA SITIO WEB:
[HTTPS://WWW.TESLA.COM/ES_MX/AUTOPILOT](https://www.tesla.com/es_mx/autopilot)

[13] JAKUB MOTYKA. (06/02/2016). DASHCAM: QUÉ SON, PARA QUÉ SIRVEN Y CUÁLES SON LAS MEJORES. 06/02/2016, DE COMPUTER HOY SITIO WEB:
[HTTPS://COMPUTERHOY.COM/LISTAS/APPS/DASHCAM-QUE-SON-QUE-SIRVEN-CUALES-SON-MEJORES-39591](https://computerhoj.com/listas/apps/dashcam-que-son-que-sirven-cuales-son-mejores-39591)

[14] IBAÑEZ. (12 DE MAYO DE 2012). CÓMO FUNCIONA EL COCHE AUTONOMO DE GOOGLE. 2012, DE MOTORPASIÓN SITIO WEB:
[HTTPS://WWW.MOTORPASION.COM/COCHES-HIBRIDOS-ALTERNATIVOS/COMO-FUNCIONA-EL-COCHE-AUTONOMO-DE-GOOGLE](https://www.motorpasion.com/coches-hibridos-alternativos/como-funciona-el-coche-autonomo-de-google)

[15] WEB, “ASISTENTE DE SEGURIDAD”, [EN LÍNEA] DISPONIBLE EN:
[HTTP://WWW.MAQUINARIAPRO.COM/ASISTENTES/ASISTENTE-DE-SEGURIDAD.HTML](http://www.maquinariapro.com/asistentes/asistente-de-seguridad.html)

[16] ALHAMA MURCIA. (25 DE OCTUBRE, 2018). CARACTERÍSTICAS DE LA NUEVA



INGENIERÍA EN SISTEMAS AUTOMOTRICES

RASPBERRY PI 4 MODEL B+. 25 DE OCTUBRE, 2018, DE PCCOMPONENTS SITIO WEB:
[HTTPS://WWW.PCCOMPONENTES.COM/CARACTERISTICAS-RASPBERRY-PI-4](https://www.pccomponentes.com/caracteristicas-raspberry-pi-4)

[17] MARCO ANTONIO. (21 OCT 2015). QUE NO TE PIERDAN LOS MEGAPÍXELES. 21 OCT 2015, DE LIFE STYLE SITIO WEB:
[HTTPS://CINCODIAS.ELPAIS.COM/CINCODIAS/2015/10/21/LIFESTYLE/1445437840_736033.HTML#:~:TEXT=AHORA%20BIEN%2C%20LO%20QUE%20DEFINE,QUE%20VER%20CON%20LOS%20MEGAP%C3%ADXELES.&TEXT=SI%20CONSULTAS%20CON%20CUALQUIER%20FOT%C3%B3GRAFO,CALIDAD%20DEPENDE%20DE%20LA%20LUZ](https://cincodias.elpais.com/cincodias/2015/10/21/lifestyle/1445437840_736033.html#:~:TEXT=AHORA%20BIEN%2C%20LO%20QUE%20DEFINE,QUE%20VER%20CON%20LOS%20MEGAP%C3%ADXELES.&TEXT=SI%20CONSULTAS%20CON%20CUALQUIER%20FOT%C3%B3GRAFO,CALIDAD%20DEPENDE%20DE%20LA%20LUZ).

[18] DAVID GIRAO. (8 NOVIEMBRE 2019). QUÉ ES Y PARA QUÉ SIRVE LA CÁMARA GRAN ANGULAR DE LOS MÓVILES. 8 NOVIEMBRE 2019, DE MOVILZONA SITIO WEB:
[HTTPS://WWW.MOVILZONA.ES/2019/11/08/QUE-ES-CAMARA-GRAN-ANGULAR-MOVILES/](https://www.movilzona.es/2019/11/08/que-es-camara-gran-angular-moviles/)

[19] JOSE ALCANTARA. (6 DE JUNIO, 2015). ¿SIGNIFICADO PCB - QUÉ ES TARJETA DE CIRCUITO IMPRESO?. 6 DE JUNIO, 2015, DE ALTIUM SITIO WEB:
[HTTPS://WWW.ALTIMUM.COM/ES/SOLUTION/WHAT-IS-A-PCB](https://www.altium.com/es/solution/what-is-a-pcb)

[20] EDUARDO MARIO GINÉS. (19 DE JULIO DE 2019). SENSORES ARDUINO. 19 DE JULIO DE 2019, DE APRENDIENDO ARDUINO SITIO WEB:
[HTTPS://APRENDIENDOARDUINO.WORDPRESS.COM/2018/04/14/SENSORES-ARDUINO-3/#COMMENT-19063](https://aprendiendoarduino.wordpress.com/2018/04/14/sensores-arduino-3/#comment-19063)

[21] ESTEBAN FERREÑO. (17 DE ENERO DE 2020). ANDROID STUDIO: ESTO ES LO QUE PUEDES HACER CON ESTE PROGRAMA TUTORIALES ANDROID. 17 DE ENERO DE 2020, DE EL ANDROID LIBRE SITIO WEB:
[HTTPS://ELANDROIDELIBRE.LESPANOL.COM/2020/01/ANDROID-STUDIO-QUE-ES-Y-PARA-QUE-SE-UTILIZA.HTML](https://elandroidelibre.lespanol.com/2020/01/android-studio-que-es-y-para-que-se-utiliza.html)

[22] RICARDO CONCEPCIÓN. (9 DE DICIEMBRE DE 2019). TARJETAS DE DESARROLLO. 9 DE DICIEMBRE DE 2019, DE RICOENPCION SITIO WEB:
[HTTPS://WWW.RJCONCEPCION.COM/PODCAST/TARJETAS-DE-DESARROLLO-EPISODIO-8/#:~:TEXT=UNA%20TARJETA%20DE%20DESARROLLO%20ES,INSTRUCCIONES%20DE%20UN%20PROGRAMA%20SUMINISTRADO](https://www.rjconcepcion.com/podcast/tarjetas-de-desarrollo-episodio-8/#:~:TEXT=UNA%20TARJETA%20DE%20DESARROLLO%20ES,INSTRUCCIONES%20DE%20UN%20PROGRAMA%20SUMINISTRADO).

[23] IVAN URIARTE. (15 DE ENERO 2020). MÓDULO GSM/GPRS: LLAMAR Y ENVIAR SMS. 15 DE ENERO 2020, DE PROMETEC SITIO WEB: [HTTPS://WWW.PROMETEC.NET/GPRS-LLAMAR-ENVIAR-SMS/](https://www.prometec.net/gprs-llamar-enviar-sms/)



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS



PROYECTO
INGENIERÍA EN SISTEMAS AUTOMOTRICES

[24] ROMAN TSAREV, ILKER USTOGLU, MUSTAFA SEÇKIN DURMUŞ. (2018). ENHANCED V - MODEL. EN ENHANCED V-MODEL (11). GERMANY: RESEARCHGATE.

[21] ROCÍO GONZÁLEZ. (4 DE MARZO DEL 2021). ¿QUÉ ES KOTLIN? EL NUEVO LENGUAJE DE PROGRAMACIÓN DE ANDROID QUE NECESITAS CONOCER. 4 DE MARZO DEL 2021, DE CREHANA SITIO WEB: <HTTPS://WWW.CREHANA.COM/MX/BLOG/TECH/QUE-ES-KOTLIN/>

[25]NEW LINE. (2020). ¿QUÉ ES UN REGULADOR DE VOLTAJE?. 2020, DE NEW LINE SITIO WEB: <HTTPS://CORPNEWLINE.COM/REGULADORES.HTM>