



**UNIVERSIDAD TÉCNICA DEL NORTE**

**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

**CARRERA DE INGENIERÍA EN TELECOMUNICACIONES**

**DISEÑO DE UN PROTOTIPO DISPENSADOR AUTOMÁTICO DE PASTILLAS  
PARA PERSONAS DE LA TERCERA EDAD**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO EN TELECOMUNICACIONES**

**AUTOR:**

**GEOVANA ELIZABETH COLIMBA POZO**

**DIRECTOR:**

**MSC. JAIME ROBERTO MICHILENA CALDERÓN**

**IBARRA, 2023**



# UNIVERSIDAD TÉCNICA DEL NORTE

## BIBLIOTECA UNIVERSITARIA

### AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

#### 1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
<b>CÉDULA DE IDENTIDAD:</b>	0402031264		
<b>APELLIDOS Y NOMBRES:</b>	Colimba Pozo Geovana Elizabeth		
<b>DIRECCIÓN:</b>	Maldonado y Velasco 10-75		
<b>EMAIL:</b>	gecolimbap@utn.edu.ec		
<b>TELÉFONO FIJO:</b>		<b>TELÉFONO MÓVIL:</b>	0998330584

DATOS DE LA OBRA	
<b>TÍTULO:</b>	Diseño de un prototipo dispensador automático de pastillas para personas de la tercera edad
<b>AUTOR (ES):</b>	Colimba Pozo Geovana Elizabeth
<b>FECHA: DD/MM/AAAA</b>	16/11/2023
SOLO PARA TRABAJOS DE GRADO	
<b>PROGRAMA:</b>	<input checked="" type="checkbox"/> <b>PREGRADO</b> <input type="checkbox"/> <b>POSGRADO</b>
<b>TÍTULO POR EL QUE OPTA:</b>	Ingeniero en Telecomunicaciones
<b>ASESOR /DIRECTOR:</b>	MSC. Jaime Michilena

## 2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 16 días del mes de noviembre de 2023.

**EL AUTOR:**



Geovana Elizabeth Colimba Pozo



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

**CERTIFICACIÓN**

MAGISTER JAIME MICHILENA, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN CERTIFICA:

Que el presente trabajo de Titulación DISEÑO DE UN PROTOTIPO DISPENSADOR AUTOMÁTICO DE PASTILLAS PARA PERSONAS DE LA TERCERA EDAD, ha sido desarrollado por la señorita Colimba Pozo Geovana Elizabeth bajo mi supervisión.

Es todo cuanto puedo certificar en honor a la verdad.

A handwritten signature in blue ink, appearing to read "Jaime Michilena", is written over a horizontal line. The signature is stylized and somewhat abstract.

MSc. Jaime Michilena Calderón

**DIRECTOR**



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

**DEDICATORIA**

*Con mucho cariño y gratitud dedico este trabajo:*

*A mis padres, quienes siempre han sido mi guía y mi mayor inspiración. Su amor incondicional y su apoyo constante han sido mi motor en los momentos más desafiantes.*

*A mi hermana, por estar a mi lado, animándome y motivándome a alcanzar mis metas*

*A mi hermano, por desafiarme a ser mejor cada día e impulsarme a superar mis límites.*

*A mi sobrina, por ser mi fuente de inspiración y felicidad, y por recordarme la importancia de perseguir mis sueños*

*Colimba Pozo Geovana Elizabeth*



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

**AGRADECIMIENTO**

*Agradezco a mis padres, quienes me han mostrado a través de su ejemplo el valor del trabajo arduo y la perseverancia. Gracias por creer en mí, alentarme a perseguir mis sueños y enseñarme que con esfuerzo y dedicación se pueden alcanzar grandes logros.*

*A mi director de tesis, Ing. Jaime Michilena, por su apoyo y orientación con sus valiosos conocimientos, no solo en la elaboración de este trabajo de titulación, sino a lo largo de todo este arduo pero gratificante trayecto académico. Agradezco su disposición para resolver mis dudas y motivarme constantemente a seguir superándome profesionalmente. También quiero agradecer a mi asesor, Ing. Carlos Vásquez, por su colaboración y ayuda para la culminación de este trabajo. A todos los docentes de quienes recibí sus conocimientos, de manera especial al Ing. Marcelo Zambrano, por su guía durante la etapa inicial de mi proyecto.*

*A mi familia, amigos y a todas las personas que me han brindado su apoyo incondicional.*

*Colimba Pozo Geovana Elizabeth*



## UNIVERSIDAD TÉCNICA DEL NORTE

### FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

#### RESUMEN

El presente trabajo de titulación aborda el desarrollo de un prototipo dispensador automático de pastillas para personas de la tercera edad, con el objetivo de mejorar significativamente la gestión de su medicación diaria. Este proyecto se centra en proporcionar una solución eficaz y segura que se adapte a las necesidades específicas de este grupo demográfico. Con el uso de tecnología, como el reconocimiento facial y una interfaz de usuario intuitiva a través de una aplicación móvil, se pretende simplificar el proceso de toma de medicamentos y garantizar su cumplimiento en los horarios programados.

El enfoque principal de este trabajo es el Diseño Centrado en el Usuario, para lo cual esto implica abordar una serie de fases que van desde la investigación de las necesidades y preferencias de las personas mayores, la definición de los requerimientos del sistema, la meticulosa selección del hardware y software adecuados para el dispensador y que sea lo más óptimo posible, entre las opciones elegidas se encuentra la computadora de placa única Raspberry Pi, pantalla táctil, cámara y los diferentes entornos de programación para el controlador y la plataforma de desarrollo de aplicaciones MIT App Inventor. En base a esa selección se realiza la implementación del código para el funcionamiento del prototipo, así como el diseño y programación de la aplicación móvil. Esta aplicación a su vez interactúa de forma eficiente con los servicios de una base de datos en nube de Firebase.

Al concluir el proyecto, se realizaron pruebas para validar el funcionamiento del dispositivo. Las pruebas evaluaron la capacidad del dispensador para dispensar medicamentos en el horario y la dosis correctos, así como el rendimiento del sistema de reconocimiento facial en diferentes condiciones de iluminación y la entrega de notificaciones al usuario final. Al finalizar, después de las semanas de prueba, los resultados de las pruebas confirmaron que el dispensador tiene un alto nivel de precisión en el reconocimiento y la entrega de medicamentos a los usuarios finales.





**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

**ABSTRACT**

This degree work addresses the development of an automatic pill dispenser prototype for elderly people, with the aim of significantly improving the management of their daily medication. This project focuses on providing an efficient and safe solution that adapts to the specific needs of this demographic group. With the use of technology, such as facial recognition and an intuitive user interface through a mobile application, the aim is to simplify the medication-taking process and ensure compliance with scheduled times.

The main focus of this work is the User-Centered Design, for which this involves addressing a series of phases ranging from researching the needs and preferences of the elderly, defining the requirements of the system, the meticulous selection of hardware and software suitable for the dispenser and that it is as optimal as possible, among the options chosen is the Raspberry Pi single board computer, touch screen, camera and different programming environments for the controller and the application development platform MIT App Inventor. Based on this selection, the implementation of the code for the operation of the prototype is carried out, as well as the design and programming of the mobile application. This application in turn interacts efficiently with the services of a Firebase cloud database.

At the conclusion of the project, tests were conducted to validate the device's performance. The tests evaluated the dispenser's ability to dispense medications at the correct time and dosage, as well as the performance of the facial recognition system in different lighting conditions and the delivery of notifications to the end user. After the weeks of testing,

the test results confirmed that the dispenser has a high level of accuracy in recognizing and delivering medications to end users.

## ÍNDICE DE CONTENIDOS

Capítulo I: Antecedentes.....	1
1.1. Tema .....	1
1.2. Problema .....	1
1.3. Objetivos.....	2
1.3.1. Objetivo General.....	2
1.3.2. Objetivos Específicos.....	3
1.4. Alcance .....	3
1.5. Justificación .....	5
Capítulo II: Estado del Arte .....	7
2.1. Adherencia y no Adherencia al Tratamiento .....	7
2.1.1. Factores que Influyen Sobre la Adherencia al tratamiento.....	8
2.1.2. Adultos Mayores: Adherencia al Tratamiento.....	9
2.2. Dispensadores de Medicamentos.....	10
2.2.1. Funcionamiento básico .....	11
2.2.2. Dispensadores Manuales.....	11
2.2.3. Dispensadores Automáticos.....	12
2.3. Forma farmacéutica .....	13
2.3.1. Clasificación de las Formas Farmacéuticas .....	13
2.3.2. Condiciones de conservación y almacenamiento .....	14
2.3.3. Construcción de Contenedores de Medicamentos .....	15

2.4. Redes inalámbricas .....	16
2.4.1. Redes Inalámbricas de Área Personal (WPAN) .....	16
2.4.2. Redes Inalámbricas de Área Local (WLAN).....	17
2.4.3. Redes Inalámbricas de Área Metropolitana (WMAN) .....	17
2.4.3. Redes Inalámbricas de Área Amplia (WWAN).....	18
2.5. Sistemas Operativos Móviles.....	18
2.5.1. Android .....	19
2.5.2. iOS .....	20
2.6. Aplicaciones Móviles.....	20
2.6.1. Aplicaciones Nativas .....	21
2.6.2. Aplicaciones Híbridas .....	21
2.6.3. Aplicaciones Web .....	22
2.6.4. Entorno de Desarrollo Integrado (IDE) .....	23
2.7. Reconocimiento Facial.....	24
2.7.1. Fases de un sistema de reconocimiento facial .....	24
2.7.2. Técnicas de Reconocimiento Facial.....	25
2.8. Placas de Desarrollo.....	32
2.8.1. Microcontroladores .....	33
2.8.2. Placas SoC System-on-Chip .....	33
2.8.3. Computadoras de Placa Única (SBC) .....	34
2.8.4. Características de las Placas de Desarrollo .....	34

2.9. Metodología.....	35
2.9.1. Diseño Centrado en el Usuario .....	35
Capítulo III: Diseño del Sistema.....	38
3.1. Análisis de la Situación Actual .....	38
3.2. Requerimientos del Sistema.....	40
3.2.1. Stakeholders.....	41
3.2.2. Requerimientos de Stakeholders.....	41
3.2.3. Requerimientos del Sistema.....	43
3.2.4. Requerimientos de Arquitectura .....	44
3.3. Elección de Hardware y Software.....	47
3.3.1. Hardware.....	47
3.3.2. Software .....	52
3.4. Diseño del sistema .....	55
3.4.1. Arquitectura del Sistema.....	55
3.4.2. Diagrama del Flujo del Funcionamiento del Sistema .....	57
3.5. Desarrollo del Sistema .....	59
3.5.1. Subsistema de Dispensación .....	59
3.5.2. Almacenamiento en la Nube.....	70
3.5.3. Subsistema de Seguridad .....	73
3.5.4. Subsistema de Reconocimiento .....	75
3.5.5. Programación de Alertas.....	85

3.5.6. Programación de la Interfaz del Dispensador .....	90
3.5.7. Diseño de Aplicación Android.....	91
3.5.8. Diseño de Estructura .....	99
Capítulo IV: Pruebas de Funcionamiento .....	103
4.1. Cronograma de Pruebas de Funcionamiento del Sistema.....	103
4.2. Prueba de Registro y Sincronización de la Aplicación Móvil .....	105
4.3. Prueba de Gestión de Medicamentos Desde la Aplicación Móvil.....	107
4.3.1. Agregar Nuevo Medicamento.....	107
4.3.3. Verificar Notificaciones Generadas .....	108
4.4. Tiempo de Respuesta del Dispensador de Pastillas .....	109
4.4.1. Tiempo de Entrega de Medicamento .....	109
4.4.2. Tiempo de Entrega de Notificaciones.....	112
4.4.3. Precisión en la Dispensación .....	113
4.5. Prueba de Usabilidad y Experiencia de Usuario.....	114
4.5.1. Prueba Usabilidad: Caso 1 .....	115
4.5.2. Prueba Usabilidad: Caso 2 .....	118
4.5.3. Prueba Usabilidad: Caso 3 .....	120
4.6. Resultados .....	122
4.6.1. Resultados Cuantitativos.....	122
4.6.2. Resultados Cualitativos.....	123
Conclusiones y Recomendaciones .....	124

Conclusiones .....	124
Recomendaciones .....	126
Bibliografía .....	127
Anexos .....	133

## ÍNDICE DE FIGURAS

Figura 1 Funcionamiento básico de un dispensador .....	11
Figura 2 Formas farmacéuticas según su estado físico .....	14
Figura 3 Mapa Mundial de iOS y Android .....	19
Figura 4 Estructura de un sistema de reconocimiento facial .....	25
Figura 5 Técnicas de reconocimiento facial .....	25
Figura 6 Las cinco primeras caras propias obtenidas de la base de datos ORL .....	27
Figura 7 Ejemplo de reducción dimensional al aplicar el análisis de componentes principales (PCA).....	28
Figura 8 LBP sobre imágenes con diferente intensidad.....	30
Figura 9 Cálculo del LBP básico .....	30
Figura 10 Red Neuronal Convolutiva Típica .....	31
Figura 11 Esquema de las fases del descriptor HOG.....	32
Figura 12 Etapas del Diseño Centrado en el Usuario .....	36
Figura 13 Arquitectura del sistema .....	57
Figura 14 Flujograma del sistema.....	58
Figura 15 Estructura del servomotor.....	60
Figura 16 Ciclo de trabajo del servomotor .....	60
Figura 17 Disposición de 40 pines GPIO en Raspberry Pi.....	61
Figura 18 Conexiones del subsistema de dispensación .....	63

Figura 19 Extracción de los datos almacenados en cada compartimiento.....	64
Figura 20 Conversión de los valores a enteros .....	65
Figura 21 Extraer la hora actual del sistema .....	66
Figura 22 Encontrar coincidencias entre la hora actual y la hora programada .....	66
Figura 23 Función para activar servomotores.....	67
Figura 24 Diagrama de flujo de programación de los compartimientos.....	68
Figura 25 Programación de lo servomotores .....	69
Figura 26 Diagrama de bloques del funcionamiento de los servomotores .....	70
Figura 27 Almacenamiento en la nube .....	71
Figura 28 Configuración de la conexión Raspberry Pi - Firebase.....	72
Figura 29 Configuración de la conexión de la aplicación móvil – Firebase.....	73
Figura 30 Diagrama de Subsistema de Seguridad .....	74
Figura 31 Programación del subsistema de seguridad.....	75
Figura 32 Diagrama general del proceso de reconocimiento facial.....	76
Figura 33 Conexiones del subsistema de reconocimiento .....	77
Figura 34 Conexión pantalla – Raspberry Pi .....	78
Figura 35 Agregar nuevo usuario a la base de datos de imágenes.....	79
Figura 36 Detección del rostro.....	80
Figura 37 Conjunto de imágenes del usuario.....	81
Figura 38 Flujograma de la programación para obtener los datos de entrenamiento ..	81
Figura 39 Procesamiento de las imágenes correspondiente al usuario de prueba .....	82
Figura 40 Entrenamiento del modelo.....	83
Figura 41 Diagrama de flujo del algoritmo de entrenamiento .....	84
Figura 42 Algoritmo de reconocimiento en ejecución.....	85
Figura 43 Iniciar conversación con BotFather.....	86



Figura 44 Creación del nuevo bot.....	87
Figura 45 Nombre y nombre de usuario para el bot. ....	88
Figura 46 Programación de notificaciones .....	89
Figura 47 Interfaz para mostrar el estado de los compartimientos .....	90
Figura 48 Acciones del botón registrar usuario .....	91
Figura 49 Diagrama de flujo de la ventana de inicio y registro.....	92
Figura 50 Diseño de la ventana inicio de sesión.....	93
Figura 51 Programación de la ventana registro .....	94
Figura 52 Diagrama de flujo del funcionamiento de la ventana registro de alarmas ..	95
Figura 53 Diseño de la ventana registro de alarmas .....	96
Figura 54 Diseño de la ventana alarmas programadas.....	97
Figura 55 Diagrama de flujo del funcionamiento del modo viaje .....	98
Figura 56 Diseño de la ventana del modo viaje.....	99
Figura 57 Estructura del dispensador de pastillas.....	100
Figura 58 Estructura del contenedor de tabletas .....	101
Figura 59 Estructura en acrílico.....	102
Figura 60 Prototipo funcional .....	102
Figura 61 Registro de usuario.....	106
Figura 62 Inicio de sesión del usuarioprueba .....	107
Figura 63 Agregar nuevo medicamento.....	108
Figura 64 Alertas generadas a través del bot de Telegram .....	109
Figura 65 Interacción del usuario 1 con el dispensador.....	116
Figura 66 Interacción del usuario 2 con el dispensador.....	119
Figura 67 Interacción de usuario 3 con el dispensador.....	121

## ÍNDICE DE TABLAS

Tabla 1 Dispensadores comerciales de medicamentos .....	12
Tabla 2 Tecnologías de las WPAN.....	16
Tabla 3 Estándares de 802.11 .....	17
Tabla 4 Comparación de los tipos de aplicaciones móviles: características.....	22
Tabla 5 Nomenclatura de los requerimientos .....	41
Tabla 6 Stakeholders del proyecto.....	41
Tabla 7 Requerimientos de stakeholders .....	42
Tabla 8 Requerimientos del sistema .....	43
Tabla 9 Requerimientos de arquitectura .....	44
Tabla 10 Comparativa de placas de desarrollo .....	47
Tabla 11 Valoración para la elección de la placa de desarrollo.....	48
Tabla 12 Especificaciones técnicas de las pantallas LCD .....	49
Tabla 13 Valoración para la elección del sensor de proximidad .....	50
Tabla 14 Especificaciones técnicas cámaras para Raspberry Pi.....	51
Tabla 15 Valoración para la selección de la cámara.....	52
Tabla 16 Especificaciones técnicas de las plataformas de desarrollo Android .....	53
Tabla 17 Valoración para la selección de la plataforma de desarrollo .....	53
Tabla 18 Características técnicas de bases de datos .....	54
Tabla 19 Valoración para la selección de la base de datos.....	55
Tabla 20 Pines para el control de los servomotores.....	62
Tabla 21 Cronograma de Pruebas .....	103
Tabla 22 Tiempo de dispensación registrado.....	110
Tabla 23 Pautas para el establecimiento de un valor óptimo de desviación estándar	111

Tabla 24 Tiempo de entrega de notificaciones a través del bot de Telegram.....	112
Tabla 25 Resultados prueba de precisión.....	114
Tabla 26 Perfil del usuario 1 – Caso 1 .....	115
Tabla 27 Resultados de la interacción del usuario 1 – caso 1.....	117
Tabla 28 Perfil del usuario 2 – Caso 2.....	118
Tabla 29 Resultados de la interacción del usuario 2 – caso 2.....	119
Tabla 30 Perfil del usuario 3 – Caso 3 .....	120
Tabla 31 Resultados de la interacción del usuario 3 – caso 3.....	121
Tabla 32 Rendimiento general de las funciones del dispensador .....	122

## Capítulo I: Antecedentes

### 1.1. Tema

DISEÑO DE UN PROTOTIPO DISPENSADOR AUTOMÁTICO DE PASTILLAS PARA PERSONAS DE LA TERCERA EDAD

### 1.2. Problema

Hoy en día es usual que las personas hagan uso de medicamentos como recurso para tratar diferentes enfermedades, en su mayoría, son personas de la tercera edad, que debido al deterioro de las funciones de su organismo requieren de tratamientos que involucran la administración continua de fármacos, que deben ser suministrados en horarios y dosis específicas. Según datos del Banco Mundial, en 2021 se estimó una tasa de 784 millones de personas de 65 años o más, lo que representaba para ese entonces el 10% de la población mundial. En Ecuador, 1.3 millones de personas son adultas mayores, es decir, 8% de la población ecuatoriana. Estas cifras van en crecimiento y se encuentran relacionadas con el aumento de la esperanza de vida de las personas, como lo indica las estadísticas de (Banco Mundial, 2022), que estima la esperanza de vida de los hombres y mujeres ecuatorianas en 77 y 80 años respectivamente.

De acuerdo con (Filié Haddad et al., 2009), “la mayor parte de individuos con 75 años o más, hacen uso continuo de por lo menos tres fármacos. Los más consumidos son antihipertensivos, diuréticos, analgésicos, antiinflamatorios, ansiolíticos y vitamínicos”. En este sentido, asegurar el uso racional de los medicamentos se vuelve primordial para aliviar los síntomas o curar una enfermedad, pues sus beneficios pueden verse alterados y conllevar múltiples consecuencias, tanto clínicas como económicas por falta de cumplimiento en el tratamiento.

En un estudio realizado por (Gray et al., 2001) sobre la adherencia a la medicación en 147 participantes mayores que tomaban tres o más medicamentos, resultó en que 30.6 % de los participantes tenían un cumplimiento insuficiente y el 18,4 % de los participantes tenían un cumplimiento excesivo con al menos un medicamento. Esto se debe a que los pacientes comprenden con dificultad las explicaciones médicas relacionadas con el tratamiento pues resultan complejas y confusas. Otras veces la causa es un simple olvido (Dilla et al., 2009).

En algunos casos, los pacientes toman dosis insuficientes o excesivas de los fármacos, debido a que no se cuenta con alguien que vigile en todo momento la ingesta adecuada, lo que provoca afectaciones en su salud, aumenta la probabilidad de aparición de resistencia a los fármacos (Organización Mundial de la Salud, 2003), y en el peor de los casos debido a la vulnerabilidad de las personas adultas mayores, una pequeña diferencia en las dosis de medicamentos pueden ser tóxicas o incluso letales como lo indica Javier Velasco, miembro de la subcomisión de Revisión del Uso de los Medicamentos de la Sociedad Española de Farmacia Familiar y Comunitaria (Sefac) (Velasco, 2022).

Es por esta razón que se plantea el desarrollo de un prototipo de dispensador automatizado de medicamentos de tipo pastilla dirigido a personas de la tercera edad, cuyo objetivo es el de dispensar los fármacos en dosis y horarios adecuados. De esta forma contribuir a que el usuario lleve su tratamiento médico con la mejor la adherencia terapéutica posible y así evitar repercusiones negativas en la salud del adulto mayor.

### **1.3. Objetivos**

#### ***1.3.1. Objetivo General***

Diseñar un prototipo dispensador automático de pastillas para el suministro adecuado en horario y dosis de medicamentos a personas de la tercera edad, mediante alertas generadas desde una aplicación móvil.

### ***1.3.2. Objetivos Específicos***

- Revisar el estado del arte de los dispensadores de medicamentos disponibles en el mercado, conceptos relacionados con reconocimiento facial y las diferentes técnicas y algoritmos de reconocimiento.
- Seleccionar los elementos y mecanismos necesarios para el diseño del dispensador inteligente de pastillas con un enfoque de diseño centrado en el usuario.
- Implementar el sistema electrónico, mecánico y de reconocimiento para el suministro de los medicamentos e interconectarlo a una aplicación móvil para la gestión remota de las acciones del dispensador y la emisión de alertas.
- Realizar pruebas para verificar el correcto funcionamiento del dispositivo

### **1.4. Alcance**

Este trabajo de grado estará enfocado en el diseño y construcción de un prototipo dispensador inteligente de pastillas dirigido a personas adultas mayores que hacen uso de múltiples medicamentos. El objetivo es controlar que el adulto mayor tenga acceso al tipo y dosis adecuada de medicina, en el horario correcto. Este proyecto se desarrollará en base a un enfoque de diseño centrado en el usuario, conformado por cuatro fases principales: análisis contextual, definición de requisitos, diseño y evaluación (Garreta & Mor, 2020).

En la primera fase, se lleva a cabo una revisión bibliográfica acerca de los diferentes dispositivos presentes en el mercado y que tuvieran el propósito de facilitar la toma de medicamentos de tipo pastilla en adultos mayores. Esto se realiza con la finalidad de extraer información de los conceptos y tecnologías que mejor se adapten al desarrollo de este proyecto, además, sirve para identificar y comprender aquellas necesidades y dificultades a las que se enfrentan las personas que hacen uso de medicamentos, que aún no han sido exploradas y que pueden ser cubiertas con mayor eficiencia con el dispositivo propuesto. Además, se revisará

conceptos acerca de reconocimiento facial y las diferentes técnicas y algoritmos existentes para el reconocimiento.

En la segunda fase, se preparan todos los recursos necesarios con los que debe contar el dispositivo, se realizará un análisis y selección de los elementos y herramientas tecnológicas que brinden un mejor desempeño al sistema planteado.

Luego de recopilar toda esta información, comienza el proceso de diseño. En este sentido, el hardware del sistema estará constituido por componentes electrónicos y electromecánicos que actuarán bajo la acción de un controlador para expulsar las pastillas y dirigir todas las acciones del dispensador, dicho controlador incorporará características de comunicación WiFi para la conexión entre el dispositivo y el celular. La fabricación de la carcasa y el mecanismo de dispensación se realizará a través de impresión 3D, con el fin de generar un ambiente óptimo para la conservación de los medicamentos. El dispositivo estará conformado por seis compartimientos con capacidad para quince pastillas. Además, podrá ser usado por dos usuarios, gracias a que contará con un mecanismo de reconocimiento facial, para la dispensación, es decir, cuando llegue la hora de tomar el medicamento se generará una alarma visual y sonora a través del teléfono, al acercarse al dispensador una cámara que estará conectada al controlador, realizará la detección, análisis e identificación y se dispensará la pastilla únicamente a la persona correcta. El sistema también será capaz de controlar si la pastilla fue retirada o no del dispensador y de notificar esta acción a la persona responsable. Para complementar el sistema, se diseñará la aplicación móvil para el sistema operativo Android, que será la puerta de entrada a todas las acciones disponibles en el dispensador, el equipo también contará con una base de datos para hacer seguimiento al uso adecuado del dispositivo y la toma de los medicamentos. A través de la aplicación móvil, el usuario podrá ingresar información como: nombre de pastillas, horario de administración, la dosis que se debe administrar.

Finalmente, en la fase de pruebas se realizan diferentes testeos en un ambiente simulado, para llegar a una versión estable según los requerimientos de los usuarios finales. De ser necesario, se realizan mejoras para optimizar el dispositivo.

## **1.5. Justificación**

Con el paso de los años, las personas mayores se vuelven más susceptibles a contraer enfermedades, esto a su vez conlleva a un aumento en el consumo de fármacos. De acuerdo con la Organización Mundial de la salud, entre 65 y 90% de los ancianos consumen algún medicamento, de esa cifra, entre un 25 y 50% cometen errores al momento de ingerir sus medicinas (Valenzuela, 2018). Así mismo, en un estudio realizado en tres centros de atención primaria de salud de Quito, se encontró que “1 de cada 4 pacientes (25%) olvida tomar al menos 1 dosis de su tratamiento y, 2 de cada 5 (40%) no cumplen los horarios establecidos para la toma de medicación, independientemente de la patología o la duración del tratamiento” (Hernández et al., 2018)

Investigaciones señalan que incumplir el tratamiento terapéutico potencia el agravamiento de la sintomatología lo que provoca una recaída y en consecuencia un ingreso hospitalario, es así como se considera que el incumplimiento por abuso, omisión o uso errático de medicamentos es la principal causa de ingresos hospitalarios, lo que empeora la calidad de vida de la persona y a su vez, resulta en un impacto económico negativo para el paciente y los sistemas de salud (Pfizer, 2021). En los adultos mayores, los cambios a nivel físico y cognitivo derivan en problemas como pérdida de memoria lo que potencia los olvidos y confusiones involuntarias a la hora de ingerir los fármacos, sumado a la complejidad para entender tratamientos y que como resultado dificulta la toma de los medicamentos de acuerdo con las instrucciones (López-Pérez et al., 2017)..



Existen algunos métodos que buscan mejorar la adherencia terapéutica, uno de ellos es el de disponer de una persona que suministre los medicamentos en las horas indicadas, lo que demanda un gasto considerable de dinero. Otro de los métodos es el uso de pastilleros para ordenar los medicamentos en varias secciones y donde el paciente debe llevar un control del tratamiento. También existen pastilleros electrónicos con capacidad para enviar alertas al usuario indicándole que debe tomar su medicamento. Sin embargo, no cuentan con aplicaciones móviles que permitan la configuración remota del dispositivo o que faciliten realizar auditorías del cumplimiento del tratamiento por parte del paciente, además, no disponen de características que permitan que el dispositivo pueda ser usado por dos usuarios (Moscoso & Villacres, 2020).

La Ley Orgánica de las Personas Adultas Mayores en el Art. 11 inciso b establece “Es corresponsabilidad de la familia cubrir sus necesidades básicas: una adecuada nutrición, salud, desarrollo físico, psíquico, psicomotor, emocional y afectivo” (Asamblea Nacional del Ecuador, 2019).

Por tal motivo, resulta fundamental brindar una solución tecnológica que contribuirá a que los adultos mayores tomen su medicación correctamente, previniendo así posibles complicaciones en la salud derivadas de la falta de adherencia al tratamiento médico. A través de un dispositivo que advierta al usuario que ha llegado la hora de tomar su medicamento, mediante notificaciones generadas desde una aplicación móvil, y que luego asegure la entrega de este, únicamente al usuario que le corresponde a través de técnicas de reconocimiento facial para evitar que terceras personas puedan manipular las medicinas. Por otro lado, se busca que los familiares o personas responsables puedan tener un control de si el medicamento fue retirado o no del dispensador, siendo estas las principales ventajas frente a otros dispositivos existentes

## **Capítulo II: Estado del Arte**

Este capítulo se enfoca en proporcionar una revisión detallada de los principales conceptos teóricos que sirven como base para el análisis y comprensión del tema de investigación. Se aborda la temática de la adherencia y no adherencia al tratamiento, la importancia de comprender este fenómeno, su impacto en la efectividad del tratamiento, especialmente en el caso de personas de la tercera edad, y en cómo los dispensadores de medicamentos pueden ayudar a mejorar la adherencia y, por lo tanto, la efectividad del tratamiento médico. En este contexto, se realiza un análisis de los dispensadores comerciales existentes, así como de los que han sido abordados como temas de investigación en estudios previos para identificar las limitaciones y las áreas de mejora. Para complementar, se indaga sobre los diferentes entornos y lenguajes de programación disponibles para el desarrollo de aplicaciones móviles. Además, se exploran los conceptos fundamentales sobre el reconocimiento facial, incluyendo los diferentes métodos y algoritmos utilizados para el análisis y la identificación de rostros humanos, con especial atención a las tecnologías de reconocimiento facial en el diseño y construcción de dispensadores de medicamentos.

### **2.1. Adherencia y no Adherencia al Tratamiento**

El término "adherencia" se refiere a la capacidad de un paciente para seguir las recomendaciones y prescripciones médicas de manera consistente y a lo largo del tiempo, lo que puede incluir tomar medicamentos regularmente, seguir una dieta específica, hacer ejercicio y asistir a las citas médicas programadas. La falta de cumplimiento puede tener efectos graves en la salud del paciente, incluida la progresión de su enfermedad y una disminución de su calidad de vida.

En países desarrollados, aproximadamente la mitad de los pacientes con enfermedades crónicas cumplen con el tratamiento prescrito, mientras que la tasa de incumplimiento varía

ampliamente dependiendo del tipo de enfermedad. En contraste, en los países en desarrollo, la tasa de adherencia al tratamiento es significativamente menor (Ortega et al., 2018).

En Ecuador, en un reciente estudio llevado a cabo en tres centros de atención primaria de salud en Quito, se descubrió que el 25% de los pacientes olvidaba tomar al menos una dosis de su tratamiento, mientras que el 40% no cumplía con los horarios establecidos para la toma de medicación, sin importar la patología o la duración del tratamiento (Hernández et al., 2018).

### ***2.1.1. Factores que Influyen Sobre la Adherencia al tratamiento***

La OMS define cinco factores que influyen sobre la adherencia al tratamiento: socioeconómicos, relacionados con el sistema de asistencia sanitaria, relacionados con la enfermedad, relacionados con el tratamiento y relacionados con el paciente.

#### **2.1.1.1. Socioeconómicos**

En países en vías de desarrollo, la adherencia al tratamiento puede verse afectada por el nivel socioeconómico bajo, lo que hace que los pacientes deban elegir entre atender sus necesidades de tratamiento y las demandas de otros miembros de la familia. Además, hay varios factores que influyen en la adherencia, como la pobreza, el analfabetismo, la falta de apoyo social, las condiciones de vida inestables, los altos costos de transporte y medicamentos, entre otros.

#### **2.1.1.2. Relacionados con el Sistema de Asistencia Sanitaria**

Los servicios de salud poco accesibles o mal equipados dificultan la atención médica oportuna y de calidad. El retraso en la entrega de medicamentos y errores en la prescripción afectan la capacidad de los pacientes para seguir el tratamiento, además, la falta de capacitación del personal médico puede generar problemas de comunicación entre el paciente y el médico, lo que afecta la comprensión del tratamiento y a su vez trae consecuencias graves para su salud.

#### **2.1.1.3. Relacionados con la Enfermedad**

Los factores vinculados con la enfermedad que el paciente experimenta dependen de la gravedad de los síntomas, si la enfermedad es grave, es mucho más probable que el paciente siga de forma rigurosa el tratamiento, en cambio, si la enfermedad es leve o los síntomas desaparecen rápidamente, es posible que el paciente descuide el tratamiento. Los efectos secundarios del medicamento también pueden ser un problema, especialmente si son graves o incómodos para el paciente, este problema podría incrementar a medida que el tratamiento se extienda o prolongue, lo que puede desmotivar al paciente (OMS, 2004).

#### **2.1.1.4. Relacionados con el Tratamiento**

Explicaciones complejas y confusas sobre el régimen médico, tratamientos extensos y que involucran múltiples medicamentos, cambios frecuentes en el tratamiento son algunos de los factores que afectan negativamente la adherencia (Pagès-Puigdemont & Valverde-Merino, 2018).

#### **2.1.1.5. Relacionados con el Sistema de Asistencia Sanitaria**

Finalmente, aquellos pacientes que puedan comprender mejor su enfermedad y su tratamiento son más propensos a seguirlo rigurosamente. Otra de las implicaciones es la salud mental del paciente, pues aquellos pacientes que sufren de depresión, ansiedad o cualquier otro problema que debilite su estado emocional, pueden tener dificultados para seguir el tratamiento médico de forma rigurosa. Por esta razón es importante que los médicos consideren estos factores al prescribir el tratamiento y proporcionen información y apoyo a los pacientes para fomentar la adherencia terapéutica.

#### **2.1.2. Adultos Mayores: Adherencia al Tratamiento**

La adherencia al tratamiento en adultos mayores es un tema de gran importancia en la actualidad, ya que esta población es cada vez más numerosa y con mayor prevalencia de enfermedades crónicas. En esta población, la adherencia al tratamiento puede verse afectada

por diversos factores, como la polifarmacia, la presencia de múltiples enfermedades crónicas, la pérdida de memoria y el deterioro cognitivo.

En un estudio sobre la alfabetización en salud y adherencia farmacológica en adultos mayores con enfermedades crónicas, específicamente en pacientes con hipertensión arterial, se analizó una muestra de 119 sujetos y se observó que el 41.2% de ellos no cumplían con su tratamiento médico (Figueroa et al., 2020). Esto se debe a diferentes factores, tales como que la falta de adquisición del tratamiento prescrito, no respetar los horarios establecidos u olvidar tomar el medicamento.

En Ecuador, en un reciente estudio sobre la adherencia terapéutica en 25 pacientes adultos mayores diagnosticados con enfermedades crónicas, a través del test de Morisky Green, se determinó que el 64% de los adultos mayores, no eran adherentes al esquema prescrito para su enfermedad (Rincón et al., 2020).

Según un estudio publicado en la revista “Gerokomos” en 2017, la adherencia al tratamiento farmacológico en adultos mayores diabéticos tipo 2 es baja. De una población de estudio conformada por 211 adultos mayores, el 48,28% no seguía su tratamiento de manera adecuada (Bello & Montoya, 2017). Este resultado sugiere la necesidad de implementar medidas para mejorar la adherencia al tratamiento de un grupo vulnerable como lo son las personas de la tercera edad y más aun considerando los riesgos asociados a la diabetes tipo 2.

## **2.2. Dispensadores de Medicamentos**

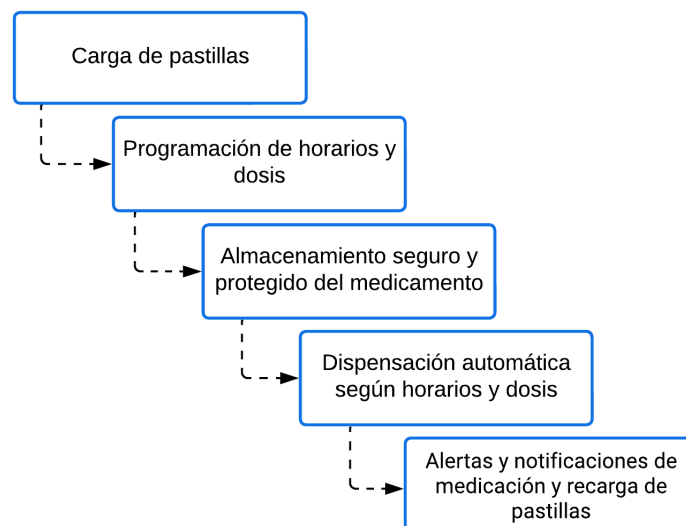
El término "dispensador de medicamentos" se refiere a un dispositivo capaz de almacenar y dispensar medicamentos. Diseñado para ayudar a las personas a tomar sus medicinas de manera efectiva y segura (Barreto, 2019). Los dispensadores de medicamentos son especialmente útiles para personas de la tercera edad, personas con enfermedades crónicas y personas que necesitan tomar varios medicamentos durante el día.

### 2.2.1. Funcionamiento básico

Los dispensadores de pastillas operan almacenando y suministrando una cantidad fija de pastillas o comprimidos en función de la hora y fecha programadas para la dosis correspondiente. En los dispensadores manuales, el usuario carga las pastillas en compartimentos separados según la hora de la dosis y, cuando llega la hora programada, las dispensa manualmente. Por otro lado, los dispensadores automáticos tienen un mecanismo programado que dispensa las pastillas en el momento indicado de forma automática. El funcionamiento de este mecanismo se puede observar en la Figura 1.

#### Figura 1

##### *Funcionamiento básico de un dispensador*



*Nota.* Adaptado de (Barreto, 2019).

### 2.2.2. Dispensadores Manuales

Los dispensadores manuales de medicamentos son dispositivos diseñados para ayudar a las personas a administrar sus medicamentos de manera más fácil y precisa. Estos dispensadores suelen tener varias secciones, cada una etiquetada con el nombre de un medicamento y el momento del día en que se debe tomar. Para utilizar un dispensador manual de medicamentos, la persona simplemente tiene que abrir la sección correspondiente al

medicamento que necesita tomar y retirar la cantidad correcta de pastillas o cápsulas. Estos dispositivos pueden ser muy útiles para personas mayores, personas con problemas de memoria o discapacidades físicas, y cualquier persona que necesite tomar varios medicamentos diferentes en momentos específicos del día.

### 2.2.3. *Dispensadores Automáticos*

En la actualidad, hay una gran variedad de dispensadores de medicamentos disponibles en el mercado. Desde estructuras con varios compartimientos en su interior que deben ser llenos manualmente y etiquetados de acuerdo momento del día para la toma del medicamento, hasta dispensadores más avanzados, automáticos e inteligentes, que ya no solo cumplen con la función de almacenar y organizar los medicamentos, sino también incluyen características y herramientas que permiten el envío de alertas o notificaciones dirigidos al paciente (Nunes, 2019). Además, permite que un familiar, cuidador o profesional de la salud, pueda realizar un seguimiento remoto de la medicación diaria de un paciente. En la Tabla 1, se evalúan tres de los dispensadores automáticos más populares, en términos de número de compartimientos, tipos de medicamentos que puede almacenar, tipo de recordatorio, costo inicial, costo de mantenimiento, entre otros.

**Tabla 1**

*Dispensadores comerciales de medicamentos*

<b>Marca/Modelo</b>	<b>MedMinder</b>	<b>Hero</b>	<b>Pria</b>
<b>Lanzamiento</b>	2015	2018	2019
<b>Capacidad de almacenamiento</b>	Hasta 28 días de suministro	Hasta 90 días de suministro	Hasta 28 días de suministro
<b>Número de compartimientos</b>	28	10	28
<b>Tipos de medicamentos que puede almacenar</b>	Cápsulas, comprimidos, tabletas	Cápsulas, comprimidos, tabletas	Cápsulas, comprimidos y tabletas

<b>Opciones de programación y configuración de la dosificación</b>	Permite programar horarios y dosis a intervalos de hasta 30 minutos	Permite programar horarios y dosis a través de una aplicación móvil a intervalos de hasta 30 minutos	Permite programar horarios y dosis a través de una aplicación móvil. a intervalos de hasta 15 minutos
<b>Tipo de recordatorio</b>	Sonoras/visuales. Envío de correo y mensajes de texto a los contactos de emergencia	Sonoras/visuales y notificaciones a los contactos de emergencia a través de la aplicación	Sonoras/visuales y notificaciones a los contactos de emergencia a través de la aplicación
<b>Costo inicial y costo de mantenimiento</b>	\$75 por mes con un compromiso de 3 años	Costo inicial de \$ 99.99, más tarifas mensuales a partir de \$ 44.99	Costo inicial de \$ 299.99, suscripción mensual de \$ 9,99
<b>¿Qué se necesita para su uso?</b>	Conexiones eléctricas	Teléfono inteligente, acceso a WiFi, aplicación Hero, suscripción a Hero	Teléfono inteligente, acceso a WiFi, aplicación Pria, suscripción a Pria

*Nota.* Adaptado de (Cheney & Hall, 2022; Clark, 2023; MedMinder, 2023)

### 2.3. Forma farmacéutica

Se refiere a la forma física en la que se presenta un medicamento para su administración. Estas formas facilitan el suministro de los fármacos al organismo, pueden variar según la vía de administración y las características del medicamento y tiene un impacto significativo en la absorción y efectividad del medicamento en el cuerpo (Pabón & González, 2017).

#### 2.3.1. Clasificación de las Formas Farmacéuticas

Existen dos clasificaciones principales para las formas farmacéuticas: según su estado físico y según su vía de administración.

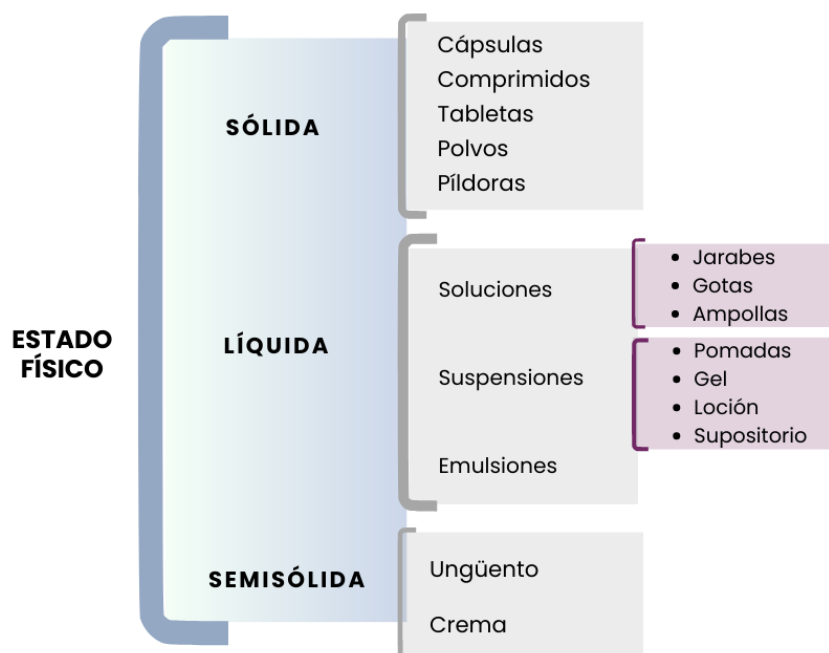
##### 2.3.1.1. Según su Estado Físico

Según su estado físico se pueden clasificar en sólidas, líquidas, semisólidas y gaseosas, véase en la Figura 2. Cada una de estas formas farmacéuticas tiene características únicas que afectan su administración y la absorción del medicamento en el cuerpo.



**Figura 2**

*Formas farmacéuticas según su estado físico*



*Nota.* Adaptado de (Pabón & González, 2017).

### 2.3.1.2. Según su vía de administración

Las vías de administración de los medicamentos se refieren a los puntos de entrada y caminos que se utilizan para que el medicamento llegue a las áreas del cuerpo donde debe actuar, ya sea en el lugar de aplicación o después de ser absorbido en el cuerpo (Pabón & González, 2017). Hay dos grupos principales de vías de administración: las indirectas o mediatas y las directas, inmediatas o parenterales.

### 2.3.2. Condiciones de conservación y almacenamiento

Los medicamentos y otros insumos para el cuidado de la salud deben conservarse de acuerdo con ciertos estándares para mantener su calidad. En general, deben almacenarse a temperatura ambiente, no superior a 30 °C, en un lugar seco y sin luz solar directa. Sin embargo, algunos productos pueden requerir condiciones especiales debido a sus propiedades físicas, químicas o terapéuticas.

Para evitar que los medicamentos se deterioren, es importante que la humedad relativa del lugar donde se almacenan no supere el 65 %. Además, el aire también puede contribuir a su degradación, ya que contiene oxígeno y humedad. Por eso, los envases deben permanecer cerrados para proteger los comprimidos de la degradación.

La luz también puede dañar los medicamentos. Por eso, los envases no suelen ser transparentes. Si lo son, suelen tener un tono opaco que filtra la luz. En algunos casos, los envases son completamente opacos para evitar que la luz entre en contacto con el medicamento (García et al., 2022).

### ***2.3.3. Construcción de Contenedores de Medicamentos***

Los materiales utilizados en la construcción de contenedores de medicamentos deben cumplir ciertos requisitos de seguridad e higiene, y varían según el tipo de medicamento y su forma farmacéutica. De acuerdo con García et al. (2022), los principales materiales empleados para la fabricación de envases de tipo farmacéutico son el vidrio, el plástico, materiales elastómeros y metales. En el caso de los envases primarios fabricados a partir de plástico, algunos de los polímeros más usados en la industria farmacéutica suelen ser el polietileno y el polipropileno.

El polietileno es un material termoplástico, blanquecino, puede ser transparente o translúcido, usualmente fabricado en láminas transparentes muy finas. Para las secciones gruesas tienden a ser translúcidas y tienen una apariencia de cera. Algunas de las propiedades que hacen del polietileno un material primario tan conveniente de utilizar para una variedad de artículos manufacturados son, su bajo peso, flexibilidad, tenacidad y alta resistencia química (García et al., 2022) .

El polipropileno es un polímero de estructura similar al polietileno. Es un plástico de alta dureza y resistencia, con color opaco y gran resistencia al calor ya que solo se ablanda a una temperatura superior a los 150 °C. Posee una mayor resistencia impermeable al vapor de

agua que el polietileno. Es altamente resistente a los golpes, aunque por otro lado posee muy baja densidad (densidad-0.9 g/ml) y se dobla muy fácilmente, resistiendo múltiples esfuerzos de fatiga. También es resistente a los productos corrosivos (García et al., 2022).

## 2.4. Redes inalámbricas

Es una agrupación, o red, de múltiples dispositivos donde los datos se envían y reciben a través de frecuencias de radio. Las redes inalámbricas generalmente incluyen alguna forma de transmisión de radio para transmitir y recibir señales inalámbricas a través de un rango específico de espectro de radiación electromagnética. Se pueden distinguir varios tipos de redes según su alcance y/o tecnología.

### 2.4.1. Redes Inalámbricas de Área Personal (WPAN)

Se basan en el estándar IEEE 802.15. Tienen un alcance limitado de hasta 10 metros. Con velocidades de transmisión inferiores a un megabit por segundo. Este tipo de redes se caracteriza por su bajo consumo de energía y son utilizadas principalmente para la comunicación entre dispositivos periféricos. Se basan en tecnologías como Bluetooth, Zigbee, UWB (Ultra-Wideband), entre otras. En la Tabla 2 se detallan algunas de las especificaciones de cada una de estas tecnologías.

**Tabla 2**

*Tecnologías de las WPAN*

<b>Tecnología</b>	<b>Bluetooth</b>	<b>Zigbee</b>	<b>UWB</b>	<b>NFC</b>
<b>Estándar</b>	802.15.1	802.15.4	802.15.3	ISO/IEC 18092
<b>Alcance</b>	≤ 100m	≤ 75 m	≤ 10 m	≤ 20 cm
<b>Velocidad máxima</b>	≤ 3 Mbps	≤ 250 Kbps	≤ 480 Mbps	≤ 424 Kbps
<b>Aplicaciones</b>	Periféricos	Domótica	Localización	Pagos móviles

*Nota.* Adaptado de (Salazar, 2016).

### 2.4.2. Redes Inalámbricas de Área Local (WLAN)

Se basan en el estándar IEEE 802.11. Proporcionan acceso inalámbrico en zonas de hasta 100 metros. Operan en diferentes bandas de frecuencia, como 2,4 GHz y 5GHz. Estas redes se basan en tecnologías como WiFi que es la más conocida, aunque existen otras, como HiperLAN2 que no recibió tanta aplicación comercial. El estándar IEEE 802.11 fue más sencillo de implementar y evoluciono con el tiempo para ofrecer velocidad de transmisión más altas. En la Tabla 3 se muestra un resumen de los diferentes estándares en términos de frecuencia, ancho del canal y velocidad de transmisión.

**Tabla 3**

*Estándares de 802.11*

<b>Protocolo</b>	<b>Frecuencia</b>	<b>Ancho del canal</b>	<b>Velocidad de transmisión</b>
<b>802.11ax</b>	2,4 o 5 GHz	20, 40, 80, 160 MHz	2,4 Gbps
<b>802.11ac</b>	5 GHz	20, 40, 80, 160 MHz	1,73 Gbps
<b>802.11n</b>	2,4 o 5 GHz	20, 40 MHz	450 Mbps
<b>802.11g</b>	2,4 GHz	20 MHz	54 Mbps
<b>802.11b</b>	2,4 GHz	20 MHz	11 Mbps
<b>802.11</b>	2,4 GHz	20 MHz	2 Mbps

*Nota.* Adaptado de (Intel, 2021).

### 2.4.3. Redes Inalámbricas de Área Metropolitana (WMAN)

Se basan en el estándar IEEE 802.16, a menudo denominado WiMAX (Worldwide Interoperability for Microwave Access). WiMAX es una tecnología de comunicaciones con arquitectura punto a multipunto orientada a proporcionar una alta velocidad de transmisión de datos a través de redes inalámbricas de área metropolitana. Esto permite que las redes inalámbricas LAN más pequeñas puedan ser interconectadas por WiMAX creando una gran WMAN. Consecuentemente, la creación de redes entre ciudades puede lograrse sin la

necesidad de cableado costoso. WiMAX opera en dos bandas de frecuencia, una mezcla de banda con licencia y banda sin licencia, de 2 GHz a 11 GHz y de 10 GHz a 66 GHz, pudiendo alcanzar velocidades de transmisión próximas a 70 Mbps en una distancia de 50 km a miles de usuarios desde una única estación base (Salazar, 2016).

#### **2.4.3. Redes Inalámbricas de Área Amplia (WWAN)**

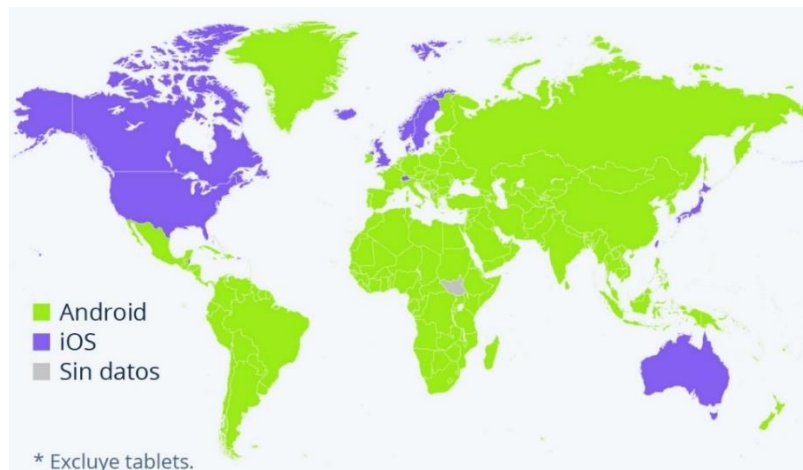
Las WWAN abarcan áreas geográficas extensas, más allá de los 50 kilómetros y se basan en varios estándares de tecnologías de comunicación móvil que han evolucionado a lo largo del tiempo, como 2G (GSM), 3G (UMTS/HSPA), 4G (LTE), y ahora 5G (NR). Estas redes ofrecen velocidades de cientos de megabits por segundo, dependiendo de la generación, y brindan conectividad a dispositivos de comunicación inalámbrica en movimiento.

### **2.5. Sistemas Operativos Móviles**

En el mercado de los sistemas operativos móviles, tanto Android como iOS son considerados como los principales líderes en la industria. Según datos de marzo de 2023, Android tiene la mayor cuota de mercado con el 71% de dispositivos móviles en todo el mundo que utilizan este sistema operativo, mientras que iOS tiene una participación de mercado del 28%, posicionándose como la segunda plataforma más utilizada a nivel mundial (Mena, 2023). La Figura 3 muestra un mapa elaborado por el portal Statista con datos tomados del servicio de análisis web StatCounter, que indica el sistema operativo más utilizado en diferentes partes del mundo. En países de Asia, Europa, Centroamérica y Latinoamérica prevalece el uso de Android, mientras que en Norteamérica y Australia prevalece el uso de iOS.

### Figura 3

#### *Mapa Mundial de iOS y Android*



*Nota.* Tomado de (Mena, 2023).

#### **2.5.1. Android**

Este sistema operativo presenta una serie de características que la han catalogado como una plataforma para móviles ampliamente utilizada. Gracias a su enfoque de código abierto, permite a los usuarios y desarrolladores acceder y personalizar el código fuente según sus necesidades. Está diseñado para ser adaptable a una variedad de dispositivos móviles con diferentes capacidades de hardware. Esto se debe a que su arquitectura está basada en componentes inspirados en Internet. Además, es posible incluir una gran cantidad de servicios como localización basada en GPS, servicios de autenticación, mensajería, pagos, publicidad, entre otros (Gironés, 2019).

La arquitectura Android se divide en cuatro capas. La capa de aplicación se compone de todas las aplicaciones que se ejecutan en un equipo Android. La capa de frameworks permite a los desarrolladores personalizar sus aplicaciones y ajustar la estructura de su sistema para que puedan aprovechar los diversos servicios que ofrecen las bibliotecas API (Sarkar et al., 2019). La capa de ejecución de Android permite que cada aplicación se ejecute en su propio entorno específico de máquina virtual Dalvik, de esta forma, diferentes aplicaciones pueden ejecutarse

en un mismo dispositivo sin sufrir interferencias. Finalmente, en el nivel más bajo de la arquitectura se encuentra el kernel o núcleo de Linux que es responsable de la gestión de procesos y recursos, así como de la seguridad y el control de acceso al sistema (Universidad Politécnica de Valencia, 2017).

### **2.5.2. iOS**

Es un sistema operativo móvil tipo Unix desarrollado por Apple que se utiliza en sus dispositivos móviles. Para desarrollar aplicaciones para iOS, se utiliza el lenguaje de programación Swift o el antiguo Objective-C, y el entorno de desarrollo integrado Xcode, que incluye herramientas para crear, depurar y perfeccionar aplicaciones (Novac et al., 2017).

La arquitectura de iOS es un conjunto de capas que proporcionan una plataforma segura y estable para la ejecución de aplicaciones. La capa de núcleo se encarga de los servicios fundamentales, contiene las funciones de bajo nivel sobre las que se basan la mayoría de las otras tecnologías. La capa de servicios proporciona servicios adicionales como la gestión de la red, la gestión de la ubicación y la conectividad de dispositivos externos. La capa de frameworks proporciona herramientas y bibliotecas de programación para desarrollar aplicaciones, incluyendo frameworks para la interfaz de usuario, gráficos, multimedia, comunicaciones con otros dispositivos y servicios. La capa Cocoa Touch es responsable de proporcionar la interfaz de usuario y la interacción táctil en dispositivos iOS. Esta capa incluye una amplia variedad de frameworks y herramientas de programación, como UIKit, Core Animation, Core Text, Core Data, entre otros (Naveen, 2023).

## **2.6. Aplicaciones Móviles**

Una aplicación móvil es un software diseñado específicamente para ser utilizado en dispositivos móviles como smartphones, tablets o smartwatches. Las aplicaciones responden a una necesidad concreta de los usuarios y permiten llevar a cabo una amplia variedad de tareas, desde el envío de mensajes, llamadas de voz y video hasta realizar compras y transacciones en

línea. Su facilidad de uso, accesibilidad e integración con otros servicios las ha convertido en una herramienta valiosa tanto para los usuarios como para las empresas (Cárdenas et al., 2019).

El uso de aplicaciones móviles para recordar la toma de medicamentos es una de las formas más comunes en que las apps se utilizan en el ámbito de la salud. A través de las aplicaciones, los usuarios pueden programar recordatorios personalizados de los medicamentos que toman y la dosis en sus dispositivos móviles. Son especialmente útiles para personas que hacen uso de varios medicamentos al día y para aquellos que tienen problemas para recordar cuando tomar sus medicamentos.

En la actualidad, existen tres tipos de aplicaciones móviles: las aplicaciones nativas, híbridas y web. La elección entre una de estas aplicaciones dependerá del proyecto y sus necesidades, como el rendimiento, requerimientos y el tiempo disponible para el desarrollo (Achulli, 2022).

### ***2.6.1. Aplicaciones Nativas***

Son aplicaciones desarrolladas específicamente para un sistema operativo en particular, como iOS o Android, utilizando un kit de desarrollo de software proporcionado por el propio sistema operativo y la plataforma, lo que les permite acceder directamente a un conjunto de códigos predefinidos conocidos como Interfaces de Programación de Aplicaciones o API, logrando así la interacción completa con las funciones y características del dispositivo y a la par brindar una experiencia de usuario más fluida (Achulli, 2022).

### ***2.6.2. Aplicaciones Híbridas***

Son aplicaciones móviles que se construyen utilizando tecnologías web como HTML, CSS y JavaScript. Estas aplicaciones se compilan y empaquetan en un contenedor web especial, lo que permite escribir una sola base de código que se puede usar en múltiples plataformas, como iOS y Android, entre otras. Esto las convierte en una excelente alternativa para los desarrolladores que buscan crear aplicaciones móviles multiplataforma (Thomas et al., 2018).



### 2.6.3 Aplicaciones Web

Estas aplicaciones se construyen para ser utilizadas en los dispositivos móviles a través de un navegador web, lo que significa que no es necesario adaptarlas a ningún sistema operativo en particular. En otras palabras, los usuarios no necesitan descargarlas ni instalarlas en su dispositivo móvil, ya que pueden acceder a ellas simplemente con una conexión a Internet, lo que supone un mínimo consumo de memoria y espacio en disco en comparación de las aplicaciones que se ejecutan localmente (Toapanta et al., 2022).

Cada tipo de aplicación tiene sus propias características y ventajas, por lo que es importante comprender las diferencias entre ellos para poder tomar decisiones informadas sobre cuál es el tipo de aplicación más adecuado para un determinado proyecto. En la Tabla 4, se presenta una comparación de las principales características de los tres tipos de aplicaciones móviles.

**Tabla 4**

*Comparación de los tipos de aplicaciones móviles: características*

<b>Característica</b>	<b>Aplicación Nativa</b>	<b>Aplicación Web</b>	<b>Aplicación Híbrida</b>
<b>Lenguaje</b>	Nativo	Nativo y web, o nativo únicamente	Web
<b>Portabilidad</b>	Bajo	Alto	Alto
<b>Gráficos avanzados</b>	Alto	Mediano	Mediano
<b>Flexible para actualización</b>	Bajo	Mediano	Alto
<b>Experiencia de instalación</b>	Alta	Alta	Mediana
<b>Uso conocimiento existente</b>	Bajo	Alto	Alto
<b>Soporte múltiple plataforma</b>	Nulo	Medio	Alto
<b>Acceso a Hardware del dispositivo</b>	Completa	Muy poco	Algunos
<b>Uso de recursos del dispositivo</b>	Alto	Bajo	Mediana

*Nota.* Tomado de (Chandi, 2017).

#### **2.6.4. Entorno de Desarrollo Integrado (IDE)**

Es un software que combina diversas herramientas y funcionalidades para facilitar el proceso de desarrollo de software. Los IDE suelen incluir un editor de código, un depurador, un compilador o intérprete, así como herramientas de gestión de proyectos, entre otras características. Estos elementos están diseñados para trabajar en conjunto y permitir a los desarrolladores crear y depurar programas de manera más eficiente y efectiva. En resumen, un IDE proporciona un conjunto completo de herramientas y funcionalidades para facilitar la creación, prueba y mantenimiento de software (Amazon, n.d.).

En el contexto de las aplicaciones móviles, los IDE son herramientas que permiten a los desarrolladores crear, probar y depurar aplicaciones móviles para dispositivos como smartphones y tablets. Algunos ejemplos de IDE populares para el desarrollo de aplicaciones móviles son Android Studio para Android, Xcode para iOS, y Visual Studio para aplicaciones móviles multiplataforma.

##### **2.6.4.1. Características comunes**

Los IDE ofrecen un conjunto de características para ayudar a los desarrolladores de software a crear y mantener código de manera más eficiente. Entre las características comunes que se encuentran en la mayoría de los IDE se incluyen un editor de código avanzado con resaltado de sintaxis y autocompletado, herramientas de depuración para detectar y corregir errores, soporte para múltiples lenguajes de programación, plantillas de código para generar código rápidamente, integración con herramientas externas y capacidad para trabajar en línea. Estas características permiten a los desarrolladores trabajar con mayor rapidez y eficiencia, reducir errores y mejorar la calidad del código (Amazon, n.d.). Cada IDE puede tener una combinación diferente de características y funcionalidades, y es importante elegir el que mejor se adapte a las necesidades y preferencias individuales de cada desarrollador.

### 2.6.4.2. Entornos Locales y en la Nube

Los IDE pueden ser clasificados en dos categorías según su lugar de ejecución: IDE locales y en la nube.

- **IDE locales.** Los desarrolladores instalan y ponen en marcha IDE locales directamente en sus equipos locales. También tienen que descargar e instalar varias bibliotecas adicionales dependiendo de sus preferencias de codificación, los requisitos del proyecto y el lenguaje de desarrollo. Mientras que las IDE locales son personalizables y no necesitan de una conexión a Internet una vez instalados, presentan varios desafíos, como el alto consumo de recursos y la ralentización de los equipos (Amazon, n.d.).
- **IDE en la nube.** Los desarrolladores utilizan IDE en la nube para escribir, editar y compilar código directamente en el navegador para prescindir de la necesidad de descargar software en sus equipos locales. Ofrecen mejor rendimiento, independencia de las plataformas y un entorno de desarrollo estandarizado (Amazon, n.d.).

## 2.7. Reconocimiento Facial

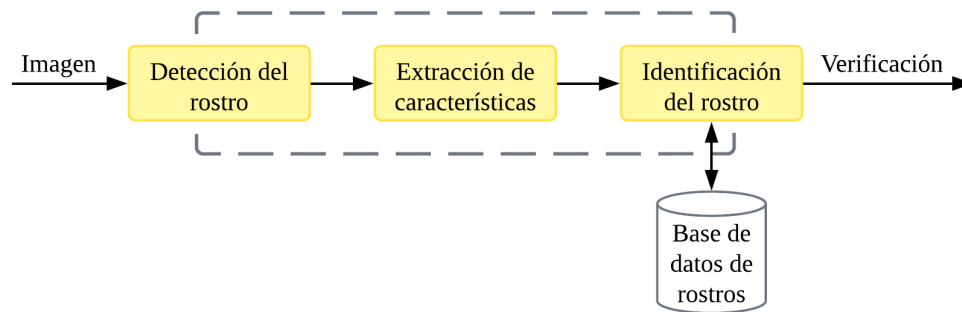
El reconocimiento facial es un método utilizado para identificar a una persona a partir de una imagen o secuencia de imágenes por medio de la detección y análisis de las características faciales únicas, como la distancia entre los ojos, la nariz, la forma de la cara, entre otros (Untuña, 2022).

### 2.7.1. Fases de un sistema de reconocimiento facial

El desarrollo de un sistema robusto de reconocimiento necesita de tres etapas básicas (Kortli et al., 2020): la detección del rostro, la extracción de características y el reconocimiento o identificación facial, como se indica en la Figura 4.

## Figura 4

### *Estructura de un sistema de reconocimiento facial*



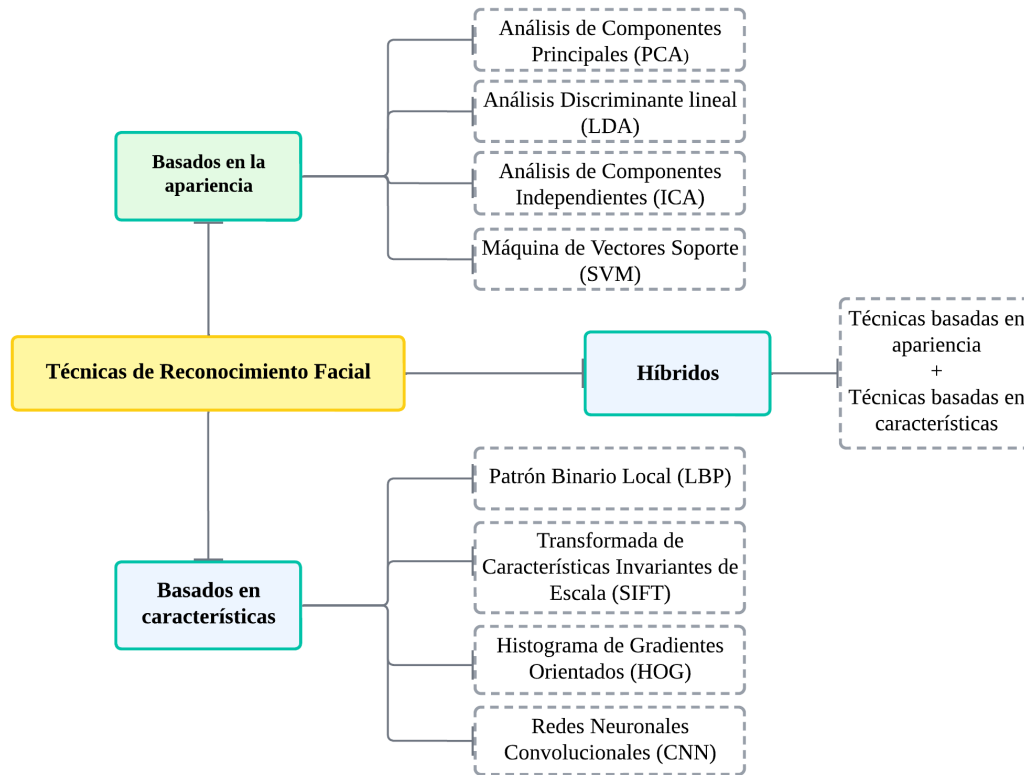
*Nota.* Adaptado de (Kortli et al., 2020).

### **2.7.2. Técnicas de Reconocimiento Facial**

Se trata de un conjunto de algoritmos y modelos matemáticos que analizan y comparan características faciales únicas de cada persona para facilitar su identificación. Según Anwarul & Dahiya (2020), existen diferentes técnicas de reconocimiento facial que se pueden clasificar en tres grupos: basados en la apariencia, en características y en modelos híbridos, como se observa en la Figura 5.

## Figura 5

Técnicas de reconocimiento facial



*Nota. Adaptado de (Anwarul & Dahiya, 2020).*

### 2.7.2.1. Técnicas Basadas en la Apariencia

Son aquellas que se enfocan en emparejar el rostro de una persona con imágenes de rostros previamente almacenados en una base de datos para su posterior identificación. El objetivo principal de este tipo de técnica es “representar la imagen facial mediante una matriz de píxeles y este vector suele convertirse en vectores de características para facilitar su tratamiento” (Kortli et al., 2020). Una de las ventajas de las técnicas de reconocimiento facial basadas en la apariencia es que se enfocan en aspectos específicos del rostro, preservando la información de las imágenes y permitiendo reconocer incluso imágenes de baja calidad. Sin embargo, tienen limitaciones debido a las variaciones faciales, lo que reduce su precisión en entornos sin restricciones. Las técnicas más populares dentro de este grupo son: análisis de componentes principales (PCA), el análisis de componentes independientes (ICA), el análisis discriminante lineal (LDA) y las redes neuronales convolucionales (Agrawal & Singh, 2015).

#### **Análisis de Componentes Principales (PCA)**

Es una técnica estadística que transforma un conjunto de datos complejos, en un nuevo y pequeño conjunto de datos o variables, mediante la identificación de los denominados componentes principales. En reconocimiento facial, se utiliza el PCA para reducir la dimensionalidad de los datos faciales, es decir, representar el rostro con un número reducido de características o también conocidas como “caras propias” como se muestra en la Figura 6 (Kortli et al., 2020).

### **Figura 6**

*Las cinco primeras caras propias obtenidas de la base de datos ORL*

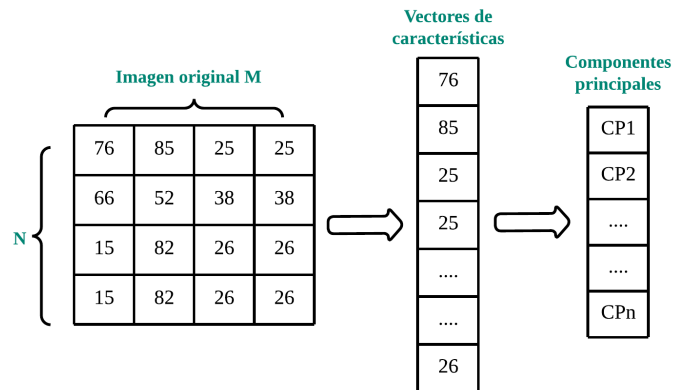


*Nota.* Tomado de (Kortli et al., 2020).

Para aplicar el PCA en las imágenes faciales, es necesario transformar las imágenes en vectores (valores en escala de grises). Luego se crea una matriz en la que las filas representan las imágenes que se desean tratar y, a su vez que serán introducidas en la base de datos (Basañez, 2018). En la Figura 7 se puede observar una representación facial simplificada que utiliza un conjunto reducido de características.

## Figura 7

*Ejemplo de reducción dimensional al aplicar el análisis de componentes principales (PCA)*



*Nota. Adaptado de (Kortli et al., 2020).*

### **Análisis de Componentes Independientes (ICA)**

Esta técnica proyecta los datos de entrada en vectores de base estadísticamente independientes (Kortli et al., 2020). En el reconocimiento facial, los datos de entrada son las imágenes del rostro. En este contexto, el ICA se emplea con el propósito de separar las características faciales en diversas componentes independientes en la medida de lo posible, con el fin de identificar rasgos faciales únicos en cada persona y analizarlos de manera independiente. De esta forma, se logra una mayor eficiencia al trabajar con imágenes que provienen de fuentes que no estén relacionadas entre sí.

### **Análisis Discriminante Lineal (LDA)**

Es una técnica de aprendizaje supervisado que ayuda a los modelos de reconocimiento facial a lograr un mejor rendimiento. LDA tiene como objetivo encontrar una transformación lineal que permita separar los datos faciales en clases específicas, reducir la dimensionalidad de la clase y, al mismo tiempo, maximizar la separabilidad entre las diferentes clases de rostros de un conjunto de datos. De esta forma se busca omitir características no deseadas y atípicas del vector final (Najafi Khanbebin & Mehrdad, 2021).

### 2.7.2.2. Técnicas Basadas en Características

En los métodos de reconocimiento facial basados en características, se utilizan partes específicas de la cara como la nariz, los ojos y la boca para encontrar coincidencias entre la imagen de la persona y las imágenes de referencia almacenadas. Estas características se utilizan para crear un modelo que se compara con los modelos almacenados en la base de datos, para determinar la identidad de la persona (Anwarul & Dahiya, 2020). Los enfoques basados en características ofrecen mayor precisión de reconocimiento que los enfoques basados en la apariencia, pero requieren grandes cantidades de datos limpios y depurados para el entrenamiento, lo que los hace más costosos en términos de memoria y recursos. En esta clasificación los más conocidos son: el reconocimiento de rostros mediante patrón binario local (LBP), redes neuronales convolucionales (CNN), histograma de gradiente orientados (HOG) (Kortli et al., 2020).

#### **Patrón Binario Local (LBP)**

Es una técnica ampliamente utilizada en el reconocimiento facial debido a su capacidad para extraer características locales que ilustran la forma y textura de una imagen facial.

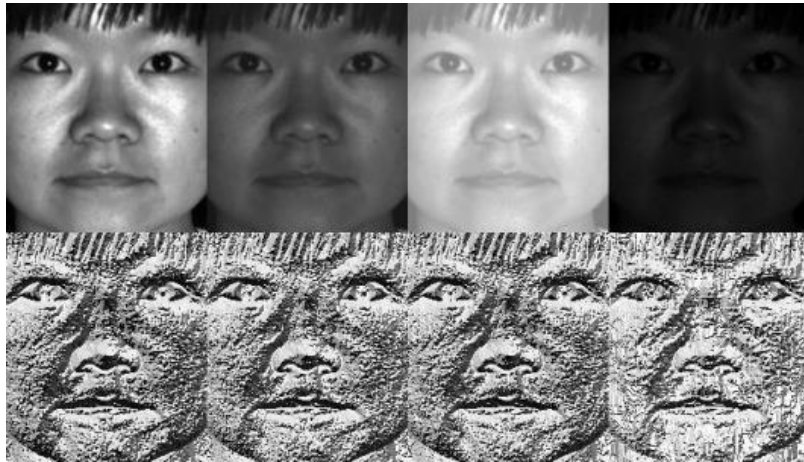
LBP es un operador de textura óptimo que marca los píxeles de la imagen mediante el umbral de cada vecindad de píxeles y determina que el resultado sea binario. El LBP funciona estableciendo un umbral entre cada píxel adyacente y el valor del píxel central. Si el valor del píxel adyacente es igual o superior al valor del píxel central, se le asigna el valor 1; en caso contrario, se le asigna el valor 0. (Ravi & Yadhukrishna, 2020).

La Figura 8 muestra la aplicación del operador de textura LBP sobre imágenes con diferente intensidad.



## Figura 8

*LBP sobre imágenes con diferente intensidad*

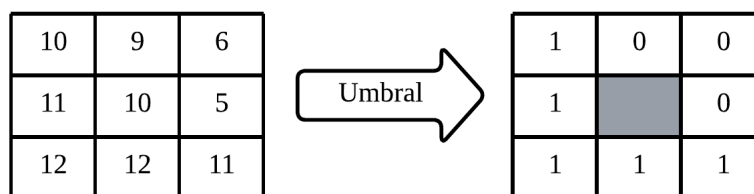


*Nota. Tomado de (Domínguez, 2017).*

La Figura 9 muestra el cálculo básico del LBP al tomar un grupo de 8 píxeles circundantes alrededor del píxel central y luego convertirlos en valores binarios mediante umbralización. Estos 8 valores binarios se utilizan como un descriptor LBP para representar esa región específica de la imagen.

## Figura 9

*Cálculo del LBP básico*



Código Binario = 10001111

*Nota. Adaptado de (Ravi & Yadhukrishna, 2020).*

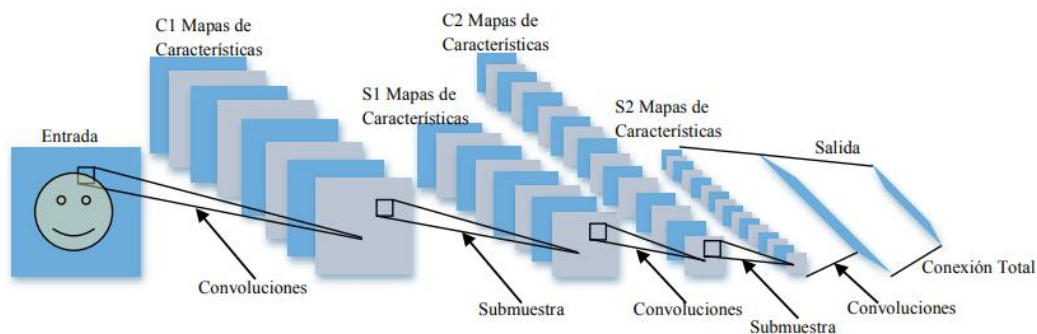
## Redes Neuronales Convolucionales (CNN)

Son un tipo de red neuronal artificial, utilizada para el procesamiento de imágenes y reconocimiento de patrones. Consisten en una serie de capas interconectadas que procesan la información de manera jerárquica. Dentro de la red neuronal convolucional (CNN), la capa

inicial es la encargada de recibir la imagen a analizar, mientras que la capa final es la responsable de proporcionar el resultado final. Además, entre estas dos capas se encuentran múltiples capas intermedias, donde se lleva a cabo el proceso de convolución, como se observa en la Figura 10. Estas capas son capaces de detectar patrones visuales cada vez más complejos, desde características simples como bordes y esquinas hasta formas y objetos más complejos. De acuerdo con (Quiroz et al., 2021), en las capas de convolución de una red neuronal, se aplican filtros a la imagen de la capa anterior. En la siguiente capa, se incluyen submuestras de la capa anterior y, opcionalmente, capas de pooling que combinan los resultados de las submuestras para reducir la complejidad de la red. La capa de conexión total conecta cada neurona de una capa con cada neurona de la siguiente. Los mapas de características ya sean de convolución o submuestreo, se obtienen al conectar cada neurona con una pequeña región de la imagen.

**Figura 10**

*Red Neuronal Convencional Típica*



*Nota.* Tomado de (Quiroz et al., 2021).

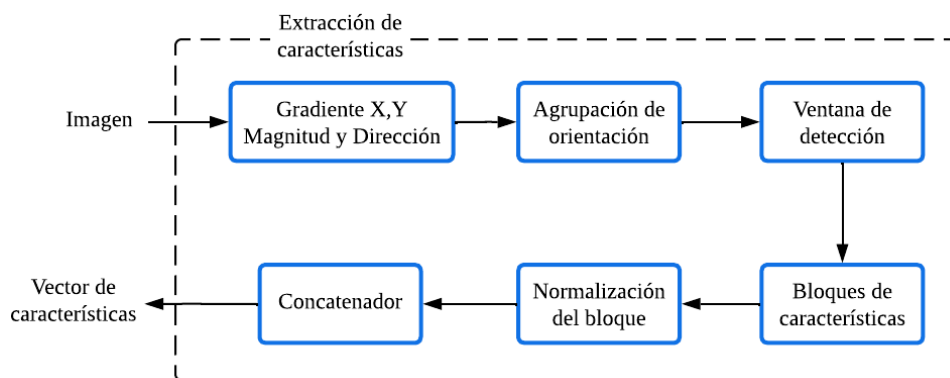
### **Histograma de Gradientes Orientados (HOG)**

Es una técnica que se utiliza para la descripción y extracción de la forma y textura de las regiones faciales. De acuerdo con (Kortli et al., 2020), se puede utilizar la técnica HOG para describir la forma de la cara analizando la distribución de la dirección de los bordes o los gradientes de intensidad de la luz en una imagen facial. El proceso consiste en dividir la imagen

en celdas o regiones pequeñas, crear un histograma de la dirección de los bordes o gradientes de cada celda, y luego combinar los histogramas de todas las celdas para obtener una característica facial única. Finalmente, se realiza un proceso de normalización y se crea el vector de características concatenando las propiedades de los bloques pertenecientes a la ventana de detección, como se observa en la Figura 11. De esta forma, se obtiene una representación precisa de la forma de la cara en la imagen.

**Figura 11**

*Esquema de las fases del descriptor HOG*



*Nota.* Adaptado de (Alvarez, 2020).

### 2.7.2.3. Técnicas Híbridas

Son una combinación de los enfoques de reconocimiento facial basados en apariencia y basados en características. Estas técnicas aprovechan las fortalezas de cada enfoque y las combinan para obtener un rendimiento superior.

## 2.8. Placas de Desarrollo

Las placas de desarrollo, también conocidas como kits de desarrollo, son una herramienta esencial para los diseñadores de sistemas. Estas placas facilitan y aceleran el desarrollo y la prueba de proyectos, ya que incorporan todo el hardware necesario para ejecutar el microcontrolador de destino.

Las placas de desarrollo suelen incluir memoria adicional, puertos de entrada y salida, LED, interruptores, un programador y otras interfaces periféricas, como UART, I2C y SPI. El hardware de estas placas ha sido probado previamente por los fabricantes, por lo que el diseñador puede centrarse en el software de la aplicación, sabiendo que el hardware funciona correctamente (Ibrahim, 2021).

### **2.8.1. Microcontroladores**

Placas basadas en una pequeña computadora, generalmente construidas con procesos CMOS. Para que el microcontrolador pueda llevar a cabo sus funciones, es necesario proporcionarle un conjunto de instrucciones precisas que indiquen cómo debe interactuar con los componentes del sistema. Estas instrucciones se codifican en un programa de software que se carga en el microcontrolador para su ejecución (Terán et al., 2022).

Se compone de cuatro bloques funcionales básicos: CPU (unidad central de procesamiento), memoria ROM (memoria solo de lectura), memoria RAM (memoria de acceso aleatorio) y periféricos de entrada y salida. Estos bloques están interconectados y trabajan en conjunto para llevar a cabo las tareas específicas que se le asignan (Terán et al., 2022). Cabe destacar que esta estructura puede variar dependiendo del modelo y del fabricante del producto.

### **2.8.2. Placas SoC System-on-Chip**

Esta placa integra en un único chip de silicio la mayoría de los componentes esenciales de un sistema electrónico. Además del procesador, el SoC generalmente incluye una GPU (unidad de procesamiento gráfico), memoria, interfaces de entrada/salida, circuitos de gestión de energía y periféricos integrados. Gracias a la integración de hardware y software en un solo chip, los SoC consumen menos energía, logran un rendimiento superior, ocupan menos espacio físico y presentan una mayor fiabilidad en comparación con los sistemas con múltiples chips (Intel, 2023).

### ***2.8.3. Computadoras de Placa Única (SBC)***

Es una placa de tamaño reducido que contienen todas las funciones de una computadora en un solo dispositivo completamente funcional, incluyendo el microprocesador, puertos de entrada/salida, memoria y otras funcionalidades que se ajustan a diferentes aplicaciones. También ofrece soporte para diversos sistemas operativos como Linux o Windows, lo que se traduce en la utilización de una variedad de lenguajes de programación. Las SBC suelen diseñadas para ser compactas, aunque su tamaño es mayor que los SoC, poseen mayor capacidad y no dependen de ranuras de expansión para agregar capacidades periféricas adicionales (Rodriguez et al., 2018).

### ***2.8.4. Características de las Placas de Desarrollo***

Para elegir una placa de desarrollo de adecuada, es importante considerar las necesidades específicas de su proyecto. Las siguientes son algunas de las características más importantes a tener en cuenta:

- **Conectividad:** La conectividad es esencial para las aplicaciones de IoT, ya que permite que los dispositivos se comuniquen con el mundo exterior. Las placas de desarrollo de IoT pueden ofrecer una variedad de opciones de conectividad, como Wi-Fi, Bluetooth, Zigbee y Ethernet.
- **Capacidades periféricas:** Las capacidades periféricas permiten a las placas de desarrollo de IoT interactuar con otros dispositivos y componentes. Algunas capacidades periféricas comunes incluyen HDMI, USB, UART, GPIO, pin-out PWM y compatibilidad con PCI/SPI.
- **Memoria:** La memoria es necesaria para almacenar datos y código. Las placas de desarrollo de IoT pueden ofrecer una variedad de opciones de memoria, como memoria flash integrada, tarjetas microSD y MiniSD.

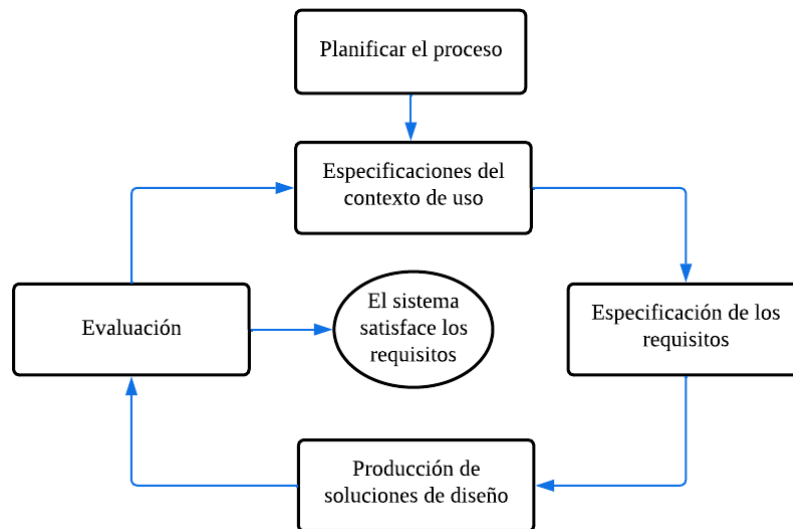
- **Potencia de procesamiento:** La potencia de procesamiento determina qué tan rápido puede funcionar la placa de desarrollo de IoT. Las placas de desarrollo de IoT pueden ofrecer una variedad de opciones de potencia de procesamiento, como microcontroladores, CPU, CPLD y FPGA.

## 2.9. Metodología

Se refiere al conjunto de procedimientos, técnicas y herramientas utilizadas para llevar a cabo la investigación científica y obtener nuevos conocimientos. Estos procedimientos y técnicas son diseñados para asegurar la validez, confiabilidad y rigor científico en la investigación y para responder a la pregunta de investigación o hipótesis planteada en el estudio. La metodología de investigación incluye varios elementos fundamentales, tales como: el sujeto, que se refiere al investigador; el objeto, que es aquello que se investiga; el medio, que está conformado por los métodos o aspectos requeridos para la investigación; y el fin, que es la solución al problema de estudio (Villanueva, 2022). Cada uno de estos elementos es esencial en la metodología de investigación y contribuye a garantizar la calidad y validez de los resultados obtenidos en el estudio.

### 2.9.1. *Diseño Centrado en el Usuario*

El diseño centrado en el usuario (DCU) es una forma de crear productos que se enfoca en las necesidades y actividades de las personas que los usarán. Esto implica comprender mejor al usuario objetivo y utilizar ese conocimiento para diseñar, evaluar y mejorar el producto en todo el proceso de diseño. El objetivo final es crear productos que sean más útiles y fáciles de usar para los usuarios (Trujillo et al., 2016).

**Figura 12***Etapas del Diseño Centrado en el Usuario*

*Nota.* Adaptado de (Navarro et al., 2019).

DCU, como muestra la Figura 12, se puede desglosar en cuatro fases interconectadas: especificaciones del contexto de uso, especificaciones de los requerimientos, producción de soluciones de diseño y la evaluación. A través de estas fases, se busca entender mejor al usuario objetivo y sus actividades, para así poder diseñar, evaluar y mejorar las propuestas de diseño en todo el proceso. Esto tiene como objetivo final crear productos más útiles y usables para los usuarios. Las fases del diseño centrado en el usuario se detallan a continuación (Navarro et al., 2019):

- **Especificar el contexto de uso.** Se enfoca identificar a los usuarios finales del producto o dispositivo, cómo utilizarán el producto en diferentes situaciones y entornos.
- **Especificar los requisitos.** Se enfoca en identificar y definir las funcionalidades y características que deben incluirse en el producto o servicio diseñado para satisfacer las necesidades y deseos de los usuarios.

- **Producir soluciones de diseño.** Conforme a los datos que se obtienen en las dos fases anteriores se lleva a cabo el diseño del producto o aplicación.
- **Evaluación.** Durante esta etapa, se lleva a cabo una evaluación de la aplicación o producto para determinar si cumple con los requisitos establecidos previamente y satisface las necesidades para las cuales fue diseñada.



### **Capítulo III: Diseño del Sistema**

En este capítulo se detalla el proceso de diseño y construcción del sistema mecánico y electrónico del prototipo, siguiendo las fases de la metodología de Diseño Centrado en el Usuario (DCU). Para complementar la primera fase, descrita en el capítulo II, se realizará un acercamiento con un especialista en geriatría y gerontología con el objetivo de conocer las limitaciones y necesidades de los adultos mayores en relación con el manejo de medicamentos. Posteriormente, se desarrollará la segunda fase, en la cual se establecen los requerimientos para la selección del hardware y software necesarios para completar la tercera fase, dónde se llevará a cabo el diseño sistema, cumpliendo con los requisitos establecidos. Con este enfoque se pretende garantizar que el prototipo final sea funcional y satisfaga las necesidades de los usuarios finales.

#### **3.1. Análisis de la Situación Actual**

En el capítulo II fue posible un primer análisis de los usuarios finales del dispositivo y de los diferentes productos disponibles en el mercado. Este apartado se centra en identificar la interacción de los adultos mayores en relación con el uso de medicamentos, tanto los problemas como las dificultades que enfrentan aquellos que están sujetos a la administración de múltiples medicamentos.

Se realizó una entrevista al Dr. Edwin Salazar, geriatra y gerontólogo. El propósito de la entrevista fue recopilar información sobre las necesidades y preocupaciones específicas de los adultos mayores con respecto al uso de medicamentos, lo que permitirá esclarecer las preguntas relacionadas con el diseño del sistema.

Durante la entrevista se abordaron temas como las enfermedades más comunes que afectan a este grupo etario, las principales barreras que impiden que las personas tomen sus medicamentos según lo prescrito y las precauciones o consideraciones que deben tenerse en

cuenta al diseñar un dispensador de pastillas para adultos mayores, tales como accesibilidad, facilidad de uso y seguridad. Las preguntas de esta entrevista se encuentran en el Anexo 1.

En concordancia con la primera pregunta, el Dr. Salazar menciona que hay muchas enfermedades que se presentan en las personas de la tercera edad. Sin embargo, las de mayor prevalencia son la diabetes, la hipertensión, cáncer y sobre todo las relacionadas con la demencia, en especial la demencia por Alzheimer.

El especialista menciona que generalmente se receta medicación en forma de pastillas o cápsulas a los adultos mayores, ya que se considera que soluciones líquidas u otras formas de medicamentos no están recomendadas para este grupo. En este sentido, los medicamentos que se recetan con mayor frecuencia para tratar enfermedades crónicas son los antihipertensivos, antidiabéticos orales como metformina y la insulina, y para las insuficiencias cardíacas, los betabloqueantes y diuréticos. Estos son solo algunos de los muchos medicamentos que se recetan a las personas de la tercera edad que padecen estas tres enfermedades.

La ansiedad y la depresión son ejemplos comunes de trastornos de salud mental que afectan a los adultos mayores. Estos trastornos pueden ser desencadenados por una serie de factores, y existen varios tratamientos para abordarlos. Entre los medicamentos que se utilizan comúnmente para tratar la ansiedad y la depresión en los adultos mayores, se encuentran los tricíclicos, la sertralina y la amitriptilina, según ha mencionado el especialista.

En el caso de enfermedades más comunes, como las infecciones respiratorias, el especialista menciona que se intenta utilizar antibióticos débiles y, en función de la respuesta del paciente, se van aumentando las dosis de acuerdo con una escala y terminología que se maneja en los hospitales públicos. En este contexto, los medicamentos más recetados son la amoxicilina, levofloxacina y claritromicina. Por otro lado, en el caso de enfermedades gastrointestinales se tienen medicamentos como ciprofloxacina y la ampicilina.

El Dr. Salazar afirma que cerca del 25% de sus pacientes que permanecen más tiempo en su consulta consumen cinco o más medicamentos. Sin embargo, en el caso de aquellos pacientes que han sido atendidos por otros especialistas o que vienen de otras instituciones de salud, más del 75% consumen cinco o más medicamentos. En este sentido, menciona que el objetivo principal del médico geriatra es reducir el consumo innecesario de medicamentos.

Entre los mayores desafíos que enfrentan los adultos mayores al tomar sus medicamentos destaca la dificultad que tienen para recordar cuándo tomarlos. Considera que aquellos pacientes que acuden a diferentes especialistas tienen mayor dificultad para cumplir con su régimen de medicamentos, ya que tienen que seguir varias recetas al mismo tiempo. Otro gran desafío son las limitaciones físicas o cognitivas. La gran mayoría de sus pacientes tienen problemas de pérdida de visión o movilidad, por lo que manipular los fármacos se vuelve una tarea difícil.

El especialista considera que los dispensadores de medicamentos son útiles para los adultos mayores, siempre y cuando se adapten e individualicen según las necesidades de cada paciente. En este sentido, menciona el caso de los pacientes que padecen de hipoacusia y que no podrían escuchar una alarma auditiva, lo que podría limitar la efectividad del dispositivo en su caso particular, por lo que hace énfasis en que el diseño se debe adecuar a las necesidades individuales de cada paciente.

### **3.2. Requerimientos del Sistema**

Dentro de esta fase se establece el conjunto de requerimientos que el prototipo debe cumplir para que sea beneficioso y se adapte a las necesidades del adulto mayor. Para poder definir estos requisitos, se utiliza la definición y terminología establecidas en el estándar ISO/IEC/IEEE:2018, como se muestra en la Tabla 5. A partir de este estándar, se definen los requerimientos de las partes interesadas (stakeholders), requerimientos del sistema y los requerimientos de arquitectura.

**Tabla 5***Nomenclatura de los requerimientos*

<b>Abreviatura</b>	<b>Descripción</b>
StRS	Especificación de requerimientos de stakeholders
SyRS	Especificación de requerimientos del sistema
ArRS	Especificación de requerimientos de arquitectura

*Nota. Adaptado de (ISO/IEC/IEEE, 2018)***3.2.1. Stakeholders**

Este término hace referencia a las personas involucradas o actores clave en el desarrollo del proyecto. La Tabla 6 contiene un listado de los stakeholders identificados, quienes tendrán influencia en la toma de decisiones durante el proceso de diseño y construcción del prototipo, incluyendo la parte electrónica, mecánica y de reconocimiento.

**Tabla 6***Stakeholders del proyecto*

<b>No</b>	<b>Denominación</b>	<b>Rol</b>
1	Adultos mayores	Usuarios directos del proyecto
2	Personas que conviven con adultos mayores	Usuarios indirectos del proyecto
3	MSc. Jaime Michilena	Director del trabajo de titulación
4	MSc. Carlos Vásquez	Asesor del trabajo de titulación
5	Geovana Colimba	Autor del trabajo de titulación

*Nota. Autoría***3.2.2. Requerimientos de Stakeholders**

Son los requisitos que deben ser satisfechos por el dispositivo para cumplir las necesidades y expectativas de los adultos mayores y otras partes interesadas. En la Tabla 7 se

detallan los requerimientos de los stakeholders, clasificados de acuerdo con su prioridad, que puede ser alta, media o baja.

**Tabla 7**

*Requerimientos de stakeholders*

<b>StRS</b>					
<b>Requerimientos de stakeholders</b>					
<b>No</b>	<b>Requerimiento</b>	<b>Prioridad</b>			<b>Relación</b>
		<b>Alta</b>	<b>Media</b>	<b>Baja</b>	
<b>Requerimientos de usuario</b>					
<b>StRS1</b>	Facilidad de uso y manipulación del dispositivo	✓			
<b>StRS2</b>	Programación de horarios y dosis	✓			
<b>StRS3</b>	Notificaciones visuales y auditivas	✓			
<b>StRS4</b>	Capacidad para almacenar diferentes medicamentos	✓			
<b>StRS5</b>	Registros y seguimiento del uso de la medicación		✓		
<b>StRS6</b>	Guía de uso del sistema	✓			
<b>Requerimientos operacionales</b>					
<b>StRS7</b>	Conexión eléctrica estable	✓			
<b>StRS8</b>	Capacidad de conectarse a una red WiFi y comunicarse a través de ella con un teléfono inteligente	✓			
<b>StRS9</b>	El smartphone en el que se va a implementar la aplicación debe contar con un sistema operativo Android 8.0 y superior			✓	

**StRS10** Operación permanente

✓

*Nota.* Autoría.

### 3.2.3. *Requerimientos del Sistema*

Son un conjunto de especificaciones que describen las funciones y características que un sistema debe poseer para satisfacer las necesidades y expectativas del usuario final. Dentro de este conjunto se encuentran los requerimientos de uso, rendimiento, interfaces y físicos. La identificación y definición de estos requerimientos debe involucrar a todos los stakeholders de esta forma se asegura que el sistema cumpla con todos los objetivos y requisitos establecidos.

En la Tabla 8 se detallan los requerimientos del sistema.

**Tabla 8**

*Requerimientos del sistema*

SyRS					
Requerimientos del sistema					
No	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
Requerimientos de uso					
<b>SyRS1</b>	Capacidad para dispensar automáticamente las pastillas	✓			
<b>SyRS2</b>	Sistema con capacidad para identificar al usuario que intenta acceder al medicamento	✓			
<b>SyRS3</b>	Programación de horarios de dispensación personalizados	✓			
<b>SyRS4</b>	Registro del cumplimiento de la medicación	✓			
<b>SyRS5</b>	Cada medicamento debe ser almacenado en un contenedor diferente	✓			

<b>Requerimientos de rendimiento</b>		
<b>SyRS6</b>	Tiempo de respuesta rápido al recibir comandos desde la aplicación móvil	✓
<b>SyRS7</b>	Alta precisión en la dispensación de las pastillas	✓
<b>SyRS8</b>	Los sensores deben estar regulados y calibrados	✓
<b>SyRS9</b>	La aplicación debe estar en ejecución continua	✓
<b>SyRS10</b>	Tiempo de espera que determina que no se ha tomado la medicación	✓
		<b>SyRS5</b>
<b>Requerimientos de interfaces</b>		
<b>SyRS11</b>	Puerto disponible para el acople de una cámara	✓
<b>SyRS12</b>	Placa con pines de entrada/salida digital	✓
<b>Requerimientos físicos</b>		
<b>SyRS13</b>	Seis contenedores individuales, cada uno con capacidad para 15 pastillas	✓
<b>SyRS14</b>	Diseño ergonómico que facilite el uso por parte de personas de la tercera edad	✓

*Nota.* Autoría

### **3.2.4. Requerimientos de Arquitectura**

En la Tabla 9 se establecen los requerimientos de la arquitectura los cuales establecen una base sólida para el diseño y construcción del sistema.

#### **Tabla 9**

*Requerimientos de arquitectura*

---

**ArRS**

---

<b>Requerimientos de arquitectura</b>					
<b>No</b>	<b>Requerimiento</b>	<b>Prioridad</b>			<b>Relación</b>
		<b>Alta</b>	<b>Media</b>	<b>Baja</b>	
<b>Requerimientos de diseño</b>					
<b>ArRS1</b>	Capacidad suficiente para albergar y mantener protegidos a los contenedores, sensor, placa de desarrollo y demás elementos tanto mecánicos como eléctricos.	✓			
<b>ArRS2</b>	Mecanismo de dispensación seguro	✓			
<b>ArRS3</b>	Resistente al uso diario y prolongado	✓			
<b>ArRS4</b>	Características para adaptarse a las necesidades específicas de cada usuario	✓			
<b>Requerimientos de software</b>					
<b>ArRS5</b>	Librerías compatibles con el sensor	✓			
<b>ArRS6</b>	Plataforma de desarrollo de código abierto		✓		
<b>ArRS7</b>	Sistema operativo de uso libre compatible con la placa	✓			
<b>ArRS8</b>	Entorno de programación intuitivo y fácil de usar		✓		
<b>ArRS9</b>	Compatibilidad con el sistema operativo Android		✓		
<b>ArRS10</b>	Compatible con el lenguaje de programación Python		✓		
<b>ArRS11</b>	Compatibilidad con librerías de visión artificial y manipulación de imágenes	✓			



<b>ArRS12</b>	Aplicación para la adquisición de imagen compatible con cámara	✓	
<b>Requerimiento de hardware</b>			
<b>ArRS13</b>	Placa de desarrollo con conectividad WiFi integrada	✓	StRS8
<b>ArRS14</b>	Placa de desarrollo debe contar con entradas y salidas digitales	✓	
<b>ArRS15</b>	Placa de desarrollo con espacio para una tarjeta de almacenamiento	✓	
<b>ArRS16</b>	Placa de desarrollo con un procesador y memoria de alta capacidad		✓
<b>ArRS17</b>	Placa con puerto de conexión para una cámara	✓	
<b>ArRS18</b>	Sensores compatibles con la placa de desarrollo utilizada	✓	
<b>ArRS19</b>	Sensores con pines de conexión analógico/digital	✓	
<b>ArRS20</b>	Sensores con voltaje de operación de 5V	✓	
<b>ArRS21</b>	Cámara con una interfaz compatible con la placa	✓	
<b>ArRS22</b>	Cámara con un conector físico compatible con la placa	✓	
<b>ArRS23</b>	Resolución de cámara no menor a 5 MP		✓
<b>ArRS24</b>	Resolución y formato compatibles con la placa		✓

*Nota.* Autoría

### 3.3. Elección de Hardware y Software

En este apartado se realiza la selección del hardware y software adecuados para garantizar el éxito del proyecto. El propósito es evaluar las características de los distintos componentes de hardware disponibles y seleccionar aquel que se ajuste y satisfaga los requerimientos definidos en la fase 2 de este proyecto. Del mismo modo, se pretende elegir el software con el que se puedan desarrollar las tareas y actividades de manera eficiente y efectiva. En este sentido, es muy importante asegurar que tanto el software como el hardware sean compatibles y puedan interactuar sin ningún inconveniente.

#### 3.3.1. Hardware

Esta selección se lleva a cabo tomando en cuenta las especificaciones de la sección de requerimientos de hardware detallados en la Tabla 9.

##### 3.3.1.1. Placa de Desarrollo

La placa es el componente principal de hardware, ya que permite la comunicación entre el dispositivo y la aplicación móvil, además del control de actuadores y sensores. En este sentido, se han considerado tres alternativas de placas, cada una de ellas con características únicas que pueden adaptarse a las necesidades del proyecto. En la Tabla 10 se exploran las especificaciones clave de estos dispositivos.

**Tabla 10**

*Comparativa de placas de desarrollo*

<b>Especificación</b>	<b>Arduino Mega</b>	<b>Raspberry Pi 4</b>	<b>ESP32</b>
<b>Procesador</b>	ATmega2560	Broadcom BCM2711, Cortex-A72 Quad-Core 1.5GHz	Dual-Core Xtensa LX6 240MHz
<b>Memoria RAM</b>	8 KB SRAM	1GB, 2GB o 4GB LPDDR4	520 KB SRAM
<b>Conexión inalámbrica</b>	No incluye	802.11 b/g/n/ac Bluetooth 5.0, BLE Gigabit Ethernet	802.11 b/g/n Bluetooth 4.2, BLE

<b>Puertos USB</b>	2 puertos 2.0	2 puertos 3.0 2 puertos 2.0	1 micro USB
<b>GPIO</b>	54	40	36
<b>Salida de video</b>	No tiene	2 micro HDMI	No tiene
<b>Salida de audio</b>	No tiene	3.5 mm jack, HDMI	No tiene
<b>Voltaje de alimentación</b>	7-12V	5V	5V

*Nota.* Adaptado de (Raspberry Pi Ltd, 2019).

Con base en la Tabla 10, se realiza una evaluación para determinar si las tres alternativas propuestas cumplen con los requerimientos establecidos para el prototipo. La valoración se dispondrá con un valor de 1 si cumple con el requerimiento y 0 si no lo cumple. La Tabla 11 muestra la valoración de los requerimientos para la elección de la placa de desarrollo.

**Tabla 11**

*Valoración para la elección de la placa de desarrollo*

<b>Selección de la placa de desarrollo</b>							
<b>Hardware</b>	<b>Requerimientos</b>						<b>Valor total</b>
	<b>StRS8</b>	<b>ArRS13</b>	<b>ArRS14</b>	<b>ArRS15</b>	<b>ArRS16</b>	<b>ArRS17</b>	
Arduino Mega 2560	0	0	1	0	0	0	1
Raspberry Pi 4	1	1	1	1	1	1	6
ESP32	1	1	1	0	1	1	5

**“1” Cumple    “0” No cumple**

*Nota.* Autoría

**Elección:** De la Tabla 11 se deduce que la computadora de placa única Raspberry Pi 4 se ajusta a los requerimientos y necesidades del proyecto ya que brinda una mayor potencia de

procesamiento y soporte para detección facial. Además, dispone de una entrada para la conexión de la cámara y de otros componentes de hardware como sensores y actuadores.

### 3.3.1.2. Pantalla Táctil

Para la selección de la pantalla se ha considerado dos alternativas disponibles en el mercado.

La Tabla 12 muestra las especificaciones técnicas en términos de voltaje de alimentación, dimensiones, distancia de detección, consumo de corriente y ángulo de detección del sensor E18-d80nk y el sensor FC-51.

**Tabla 12**

*Especificaciones técnicas de las pantallas LCD*

<b>Especificación</b>	<b>7" TFT 1024*600 LCD HDMI</b>	<b>FC-51</b>
<b>Voltaje de alimentación</b>	5 V	3,3 V – 5 V
<b>Dimensiones</b>	15cm x 9cm	Depende del fabricante
<b>Potenciómetro</b>	No	Si
<b>Consumo de corriente</b>	100 Ma	23 mA (3.3V); 43 mA (5V)
<b>Ángulo de detección</b>	35°	35°
<b>Tiempo de respuesta</b>	< 2 ms	No se especifica

*Nota.* Autoría

**Elección:** Según los datos presentados en la Tabla 12, se puede concluir que la pantalla de 7 pulgadas TFT con resolución de 1024x600 y conexión HDMI se destaca como la opción más idónea. Esto se fundamenta en sus propiedades mencionadas previamente, las cuales confirman que, gracias a sus dimensiones, satisface el requisito de permitir a una persona de la tercera edad leer con comodidad y utilizar el dispositivo de manera eficiente.

### 3.3.1.3. Sensor de proximidad

Una vez definidas las características de los dos sensores, se realiza una evaluación para determinar si cumplen con los requisitos de hardware establecidos en la Tabla 9. El sensor que cumpla con la mayor cantidad de requerimientos será considerado como el óptimo para el desarrollo de proyecto. En la Tabla 13 se muestra la valoración.

**Tabla 13**

*Valoración para la elección del sensor de proximidad*

<b>Selección del sensor de proximidad</b>				
<b>Hardware</b>	<b>Requerimientos</b>			<b>Valor total</b>
	<b>ArRS18</b>	<b>ArRS19</b>	<b>ArRS20</b>	
<b>MC-38</b>	1	1	1	3
<b>FC-51</b>	1	0	1	2
	<b>“1” Cumple</b>	<b>“0” No cumple</b>		

*Nota.* Autoría

**Elección:** En la Tabla 13 se puede observar que la valoración total de los dos sensores se puede concluir que el sensor MC-38 es óptimo debido a que cumple con más requerimientos de los cuales son tanto energéticos como funcionales para el correcto funcionamiento del prototipo.

### 3.3.1.4. Cámara

En el mercado existe una amplia variedad de cámaras enfocadas en realizar tareas de reconocimiento facial. La elección de este dispositivo depende de varios factores como la resolución, la calidad de imagen, la sensibilidad a luz y otros aspectos como la compatibilidad con la placa escogida, incluso del entorno en donde se planea poner en marcha el proyecto. Para este caso específico se han tomado en consideración tres alternativas, en la Tabla 14 se detallan las especificaciones de mayor interés y aporte para el sistema.

**Tabla 14***Especificaciones técnicas cámaras para Raspberry Pi*

<b>Especificación</b>	<b>Cámara HQ Raspberry Pi</b>	<b>Cámara Raspberry Pi v2</b>	<b>Cámara Raspberry Pi v1</b>
<b>Sensor</b>	Sony IMX477R	Sony IMX219	OmniVision OV5647
<b>Resolución</b>	12,3 MP	8 MP	5 MP
<b>Tamaño de pixel</b>	1,55 $\mu\text{m}$ x 1,55 $\mu\text{m}$	1.12 $\mu\text{m}$ x 1.12 $\mu\text{m}$	1,4 $\mu\text{m}$ x 1,4 $\mu\text{m}$
<b>Campo de visión</b>	62,2° horizontal 48,8° vertical 73,5° diagonal	62° horizontal 48° vertical 73° diagonal	54° horizontal 41° vertical 65° diagonal
<b>Enfoque</b>	Fijo	Fijo	Ajustable
<b>Apertura</b>	f/2.0	f/2.0	f/2.9
<b>Conexión</b>	CSI-2 (15 pines)	CSI (15 pines)	CSI (15 pines)
<b>Formato de imagen</b>	Imágenes fijas: JPEG y RAW	Imágenes fijas: JPEG	Imágenes fijas: JPEG
<b>Precio de mercado</b>	\$59.99	\$45	\$10,92

*Nota.* Adaptado de (Raspberry Pi, 2020).

A partir de la Tabla 14, se realiza una evaluación de las características de cada dispositivo para determinar cuál se ajusta mejor a las necesidades del sistema. A breves rasgos es posible inferir que las tres cámaras cumplen con los requerimientos establecidos. Sin embargo, hay que tomar en consideración el costo de adquisición y envío del producto, con el fin de reducir en gran medida los costos totales de construcción del prototipo. La Tabla 15 detalla la valoración realizada a las tres alternativas.

**Tabla 15***Valoración para la selección de la cámara*

<b>Selección de la cámara</b>					
<b>Hardware</b>	<b>Requerimientos</b>				<b>Valor total</b>
	<b>ArRS21</b>	<b>ArRS22</b>	<b>ArRS23</b>	<b>ArRS24</b>	
<b>Cámara HQ Raspberry Pi</b>	1	1	1	1	4
<b>Cámara Raspberry Pi v2</b>	1	1	1	1	4
<b>Cámara Raspberry Pi v1</b>	1	1	1	1	4
	<b>“1” Cumple</b>		<b>“0” No cumple</b>		

**Elección:** En este caso particular, las tres alternativas tienen características que satisfacen con todos los requerimientos del proyecto, como las prestaciones, el formato y compatibilidad con la placa elegida. Por esta razón, la selección se la hará en torno al precio del producto. Como se observa en la Tabla 15, la cámara Raspberry Pi v1 tiene el menor costo y es posible encontrarla en tiendas nacionales, por lo que su envío no representa precios elevados. A diferencia de las otras opciones en las que su valor se eleva de manera considerable. El objetivo de esto es reducir en gran medida el costo de construcción del prototipo.

### **3.3.2. Software**

Después de seleccionar el hardware necesario para la construcción del prototipo, se procede con la selección del software, tomando como punto de partida los requerimientos definidos en la Tabla 9 de este documento. Con esta premisa, se pretende garantizar la compatibilidad entre hardware y software

#### **3.3.2.1. Aplicación Móvil**

La elección de la plataforma de desarrollo de la aplicación móvil para el sistema operativo Android se realiza en torno a tres alternativas. Analizando cada una de sus

características, se determina cual es la óptima para el sistema. La Tabla 16 muestra las especificaciones y consideraciones de cada una de estas opciones en términos del lenguaje de programación, plataformas que soporta y del tipo de licencia.

**Tabla 16**

*Especificaciones técnicas de las plataformas de desarrollo Android*

<b>Especificación</b>	<b>Android Studio</b>	<b>App Inventor</b>	<b>Flutter</b>
<b>Lenguajes de programación</b>	Java, Kotlin, C++,C, XML	Java, Blockly	Dart
<b>Plataformas soportadas</b>	Android	Android	Android, iOS, Web
<b>Desarrollo multiplataforma</b>	No	No	Si
<b>Licencia</b>	Código libre	Código libre	Código libre

Una vez listadas las características más relevantes, se realiza una evaluación para determinar cuál de las opciones cumplen con los requerimientos de software que se indican en la Tabla 9. La valoración permitirá escoger la plataforma que más se ajuste a las necesidades del sistema. La Tabla 17 muestra la valoración de cada uno las alternativas propuestas.

**Tabla 17**

*Valoración para la selección de la plataforma de desarrollo*

<b>Selección de la plataforma de desarrollo Android</b>					
<b>Hardware</b>	<b>Requerimientos</b>				<b>Valor total</b>
	<b>ArRS6</b>	<b>ArRS8</b>	<b>ArRS9</b>	<b>ArRS10</b>	
<b>Android Studio</b>	1	1	1	0	3
<b>App Inventor</b>	1	1	1	1	4
<b>Flutter</b>	1	1	1	0	3

“1” Cumple      “0” No cumple



*Nota.* Autoría

**Elección:** La valoración total muestra a App Inventor como la plataforma que satisface los requerimientos de software debido a enfoque en la facilidad de uso, la programación visual mediante bloques y compatibilidad con el lenguaje de programación Java, componentes con los que se puede desarrollar una aplicación móvil adecuada para el control del dispensador.

### 3.3.2.2. Base de datos

El almacenamiento y organización de los datos es crucial para mantener los datos de la aplicación de manera persistente, por esto razón se han considerado dos alternativas, las cuales se evalúan en base a sus especificaciones técnicas. La Tabla 18 muestra una comparativa de las tres opciones en términos de arquitectura, almacenamiento, seguridad, escalabilidad, entre otros.

**Tabla 18**

*Características técnicas de bases de datos*

<b>Especificación</b>	<b>MySQL</b>	<b>Firebase</b>
<b>Arquitectura</b>	Relacional	NoSQL sincroniza los datos en tiempo real
<b>Almacenamiento en la nube</b>	No	Sí
<b>Lenguajes soportados</b>	C, C++, C#, Java, JavaScript, Perl, PHP, Python, Ruby, Visual Basic	JavaScript, Node.js, Java, Objective-C, Swift, Python, Unity
<b>Escalabilidad</b>	Escalan verticalmente (datos en un solo servidor)	Escalan horizontalmente (datos en varios servidores)
<b>Rendimiento</b>	Funcionan bien con operaciones de lectura de conjuntos pequeños	Sistemas distribuidos que reducen la carga
<b>Seguridad</b>	Control de acceso y autenticación	Autenticación y reglas de acceso personalizadas

La selección se la realiza en torno a dos diferentes arquitecturas de base de datos. En la Tabla 19 se muestra la valoración de acuerdo con el requerimiento de software satisfecho.

**Tabla 19**

*Valoración para la selección de la base de datos*

Selección de la base de datos					
Hardware	Requerimientos				Valor total
	ArRS6	ArRS8	ArRS9	ArRS10	
MySQL	1	1	1	0	3
Firestore	1	1	1	1	4

“1” Cumple    “0” No cumple

*Nota.* Autoría

**Elección:** La valoración total muestra a Firestore como la mejor alternativa de base de datos para este proyecto. La integración con Mit App Inventor permitirá gestionar los archivos y datos en la nube, sumado a al acceso eficiente de los datos sincronizados automáticamente en tiempo real, lo que a su vez permite el envío de notificaciones push al usuario.

### 3.4. Diseño del sistema

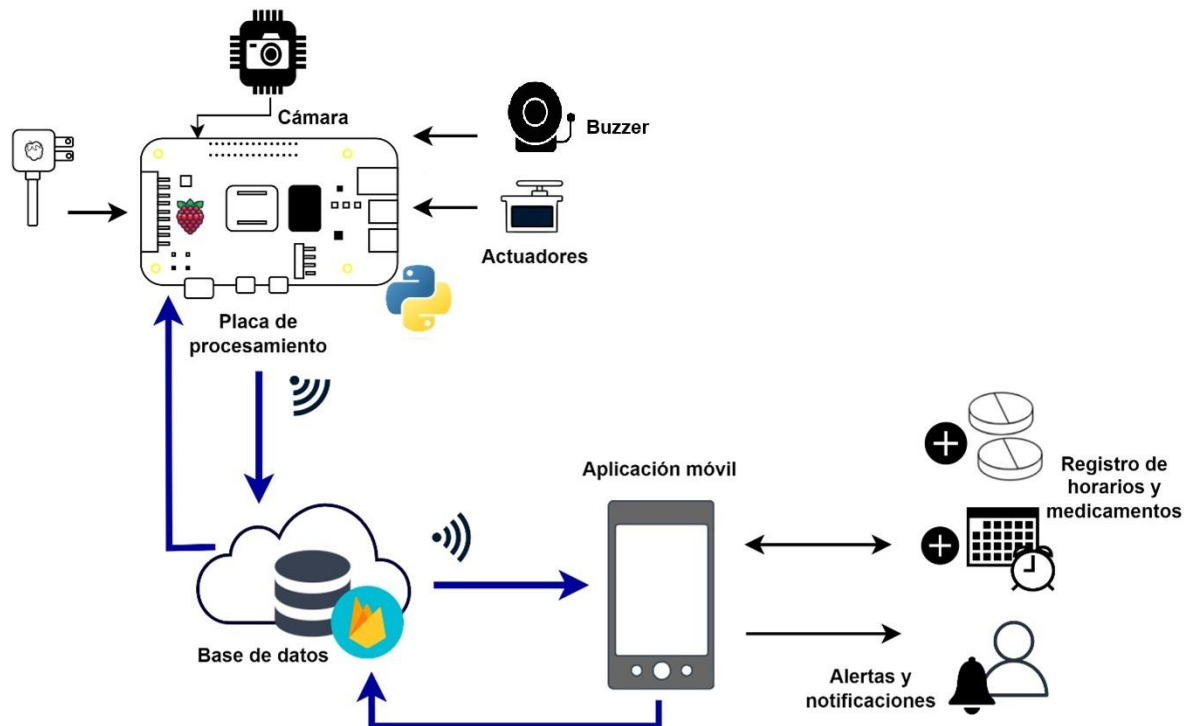
Luego de realizar el análisis de los requisitos de hardware y software, se inicia con la tercera fase del desarrollo, que implica la creación de una propuesta de diseño, tomando en consideración los requerimientos de stakeholders, sistema y arquitectura definidos en el apartado 3.2 de este documento.

#### 3.4.1. Arquitectura del Sistema

El sistema electrónico tiene como elemento central a la computadora de placa única Raspberry Pi, que tiene a su cargo el control y gestión de otros componentes de hardware como el sensor de proximidad E18-d80nk, cuya función es la de detectar la presencia del usuario y

determinar si este ha recogido la pastilla que se encuentra dispensada; los actuadores cuya función es la del control mecánico de los contenedores de pastillas para garantizar que se dispense la pastilla de forma correcta; y la cámara v1 para Raspberry Pi utilizada para la captura del rostro del usuario, acción con la que pretende controlar el acceso del usuario a su respectiva dosis.

Del lado del componente de Software, se desarrolla una aplicación móvil específica para el dispensador de pastillas. A través de esta aplicación, los usuarios pueden programar horarios y dosis personalizadas para la toma de los medicamentos y recibir los recordatorios en la hora programada. Para guardar los datos del registro de horarios, se utiliza el almacenamiento en la nube de Firebase, que permitirá guardar y sincronizar los horarios programados, al tratarse una de una base de datos en tiempo real, los datos se mantendrán actualizados. En la Figura 13 se describe la arquitectura correspondiente al diseño del prototipo de dispensador automatizado de pastillas, gestionado mediante una aplicación móvil. Esta representación permite evidenciar los procesos que deben ejecutarse para la puesta en marcha del sistema.

**Figura 13***Arquitectura del sistema*

*Nota.* Autoría

### 3.4.2. Diagrama del Flujo del Funcionamiento del Sistema

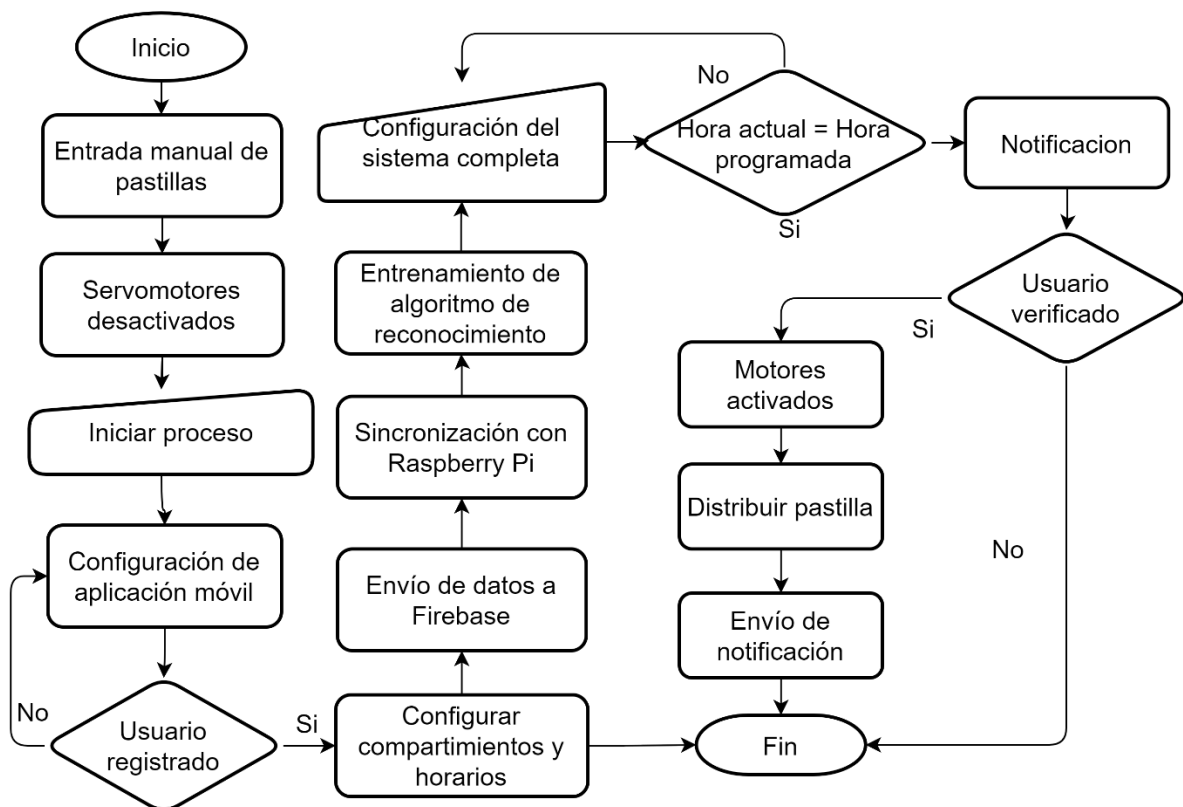
El funcionamiento del dispensador de pastillas gestionado por una aplicación móvil, radica en una serie de procesos que deben ser ejecutados de manera secuencial. Antes de empezar con la configuración del dispensador, es necesario llenar los compartimientos con las pastillas de forma manual. Luego, el usuario debe registrarse en la aplicación móvil, al momento de registrarse se le solicitará el nombre de usuario, la contraseña y una foto de su rostro, estos datos son enviados a una base de datos en Firebase. Mientras tanto, la Raspberry Pi ya se encuentra funcionando y realiza sondeos cada 60 segundos a los datos de los disponibles en la base de datos. En cuanto, descubra que ya hay información de alguno de los compartimientos, inmediatamente empieza a leer, extraer y tratar esos datos, de tal forma de que pueda almacenar en variables la hora, minutos y pueda compararlos con el horario actual

del sistema. Por otro lado, la Raspberry descarga la imagen ingresada en el registro y realizar el entrenamiento del algoritmo de reconocimiento. Una vez que este proceso este completo y se hayan registrado los datos desde la aplicación móvil el sistema estará listo para usarse.

En caso de detectar una coincidencia entre el horario actual y el horario programado desde la aplicación, se enviará una notificación al usuario para recordarle que es la hora de tomar su medicamento. En cuanto se acerque al dispensador, accionará un botón que le permitirá verificar a través de la cámara su identidad. Si se descubre que quien intenta acceder es el usuario previamente registrado, se ejecuta una secuencia que activa el servomotor correspondiente al compartimiento y la pastilla se dispensa para que el usuario pueda retirarla. Es posible observar este proceso en el flujograma de la Figura 14.

**Figura 14**

*Flujograma del sistema*



*Nota. Autoría*

### **3.5. Desarrollo del Sistema**

El desarrollo del dispensador de pastillas se ha dividido en tres subsistemas: dispensación, reconocimiento y la aplicación móvil. Estos subsistemas trabajan en conjunto para proporcionar un sistema totalmente funcional.

#### ***3.5.1. Subsistema de Dispensación***

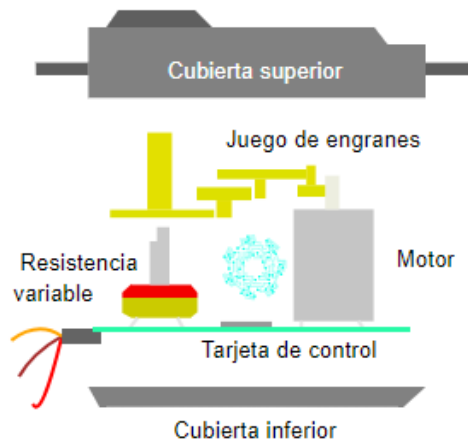
El subsistema de dispensación se encarga de la entrega precisa de los medicamentos programados. Para lograr esto, se ha implementado el uso de servos, que son dispositivos electromecánicos capaces de controlar el movimiento de manera precisa. Estos servos se activan mediante señales eléctricas y permiten que el dispensador libere la dosis correcta de medicamento en el momento adecuado.

##### **3.5.1.1. Diagrama de Conexiones del Subsistema de Dispensación**

Para este subsistema se ha requerido la acción de servomotores que controlaran los movimientos necesarios para dispensar las pastillas. Este dispositivo electromecánico cuenta con un motor que es responsable del movimiento físico, un sistema electrónico que controla el motor y garantiza que se mueva a la posición correcta y también un mecanismo de retroalimentación que conoce la posición actual del dispositivo. En la Figura 15, se logra apreciar la estructura de este elemento.

## Figura 15

### *Estructura del servomotor*



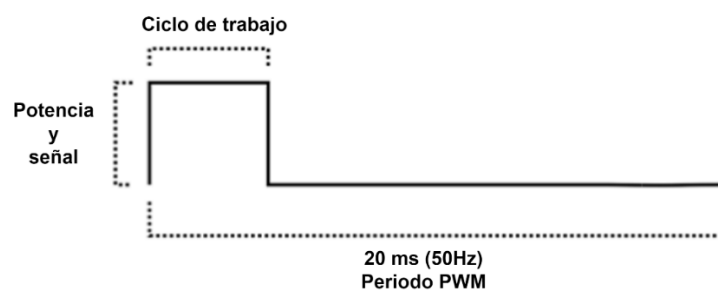
*Nota.* Extraído (MecatrónicaLATAM, 2021)

El servomotor mg995 tiene un rango de movimiento de  $120^\circ$ , lo que significa que puede girar el objeto  $60^\circ$  en cada dirección y puede detenerse en cualquier posición dentro de ese rango. Para determinar la posición deseada del eje de salida, el mg995 utiliza una señal de control llamada modulación de ancho de pulso (PWM), la duración de un pulso PWM determina el ángulo al cual debe girar.

El mg995 requiere una señal PWM con una frecuencia de 50 Hz para funcionar correctamente, por lo que, si se utiliza una frecuencia mayor o menor a este valor, provocará un mal funcionamiento del servomotor. Como se ilustra en la Figura 16, cada ciclo de la señal PWM debe tener una duración de 20 ms para una frecuencia de 50 Hz.

## Figura 16

### *Ciclo de trabajo del servomotor*

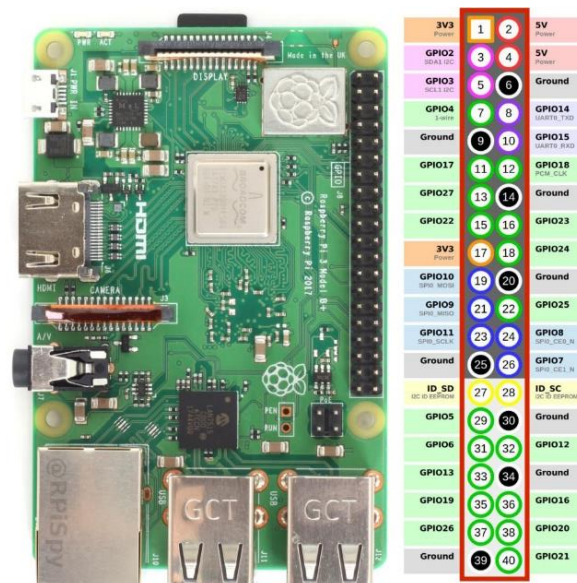


*Nota. Adaptado de (Jones, 2016)*

La Raspberry Pi tiene 40 pines GPIO, como se muestra en la Figura 17, que se pueden utilizar para controlar una variedad de dispositivos, incluyendo servomotores. Los pines GPIO son pines digitales, lo que significa que pueden enviar o recibir señales digitales, como 1 o 0. Para el control de un servomotor con una Raspberry Pi, es necesario la conexión del pin de señal a uno de los pines GPIO, en este caso la señal es una onda cuadrada que varía la posición el servomotor.

### Figura 17

*Disposición de 40 pines GPIO en Raspberry Pi*



*Nota. Tomado de (Hawkins, 2018)*

Aunque todos los pines de la Raspberry Pi se pueden utilizar para controlar los servomotores, los pines GPIO 17, 27, 22, 23, 24, 25, 8, 10 suelen ser comúnmente utilizados, los otros pines GPIO suelen tener problemas de compatibilidad con todos los modelos de servomotores disponibles en el mercado. En la Tabla 20 se muestran los pines GPIO utilizados para general señales PWM y controlar la posición y velocidad de los servomotores.



**Tabla 20***Pines para el control de los servomotores*

<b>GPIO</b>	<b>SERVO PWM</b>
GPIO 24	1
GPIO 27	2
GPIO 25	3
GPIO 22	4
GPIO 17	5
GPIO 23	6

*Nota. Autoría*

En la Figura 18 se logra apreciar la interacción de los componentes eléctricos y electromecánicos. En principio, debido a que la Raspberry Pi no puede proporcionar suficiente corriente a través de su salida de 5V para alimentar los seis servomotores, es necesario agregar una fuente de poder externa de 5V. Para este caso en específico se están utilizando servomotores que consumen 250 mA. Para calcular el consumo total se multiplican el consumo de corriente de un solo servomotor por la cantidad de servomotores.

$$\text{Consumo total de corriente} = \text{consumo de cada servo} \times \text{número de servos}$$

$$\text{Consumo total de corriente} = 250 \text{ mA} \times 6$$

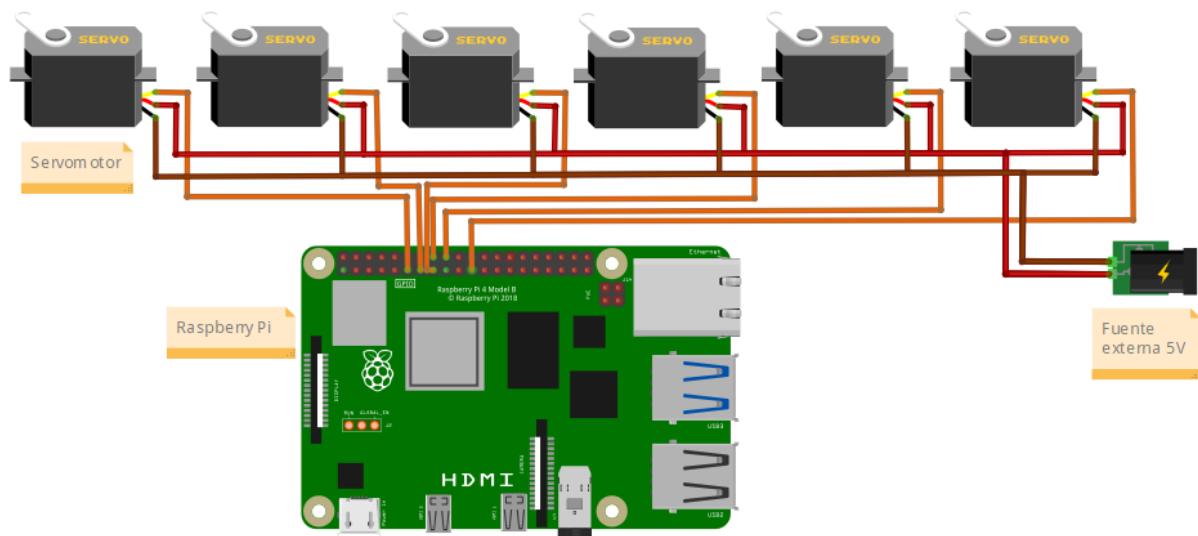
$$\text{Consumo total de corriente} = 1.5 \text{ A}$$

Tomando como premisa el resultado de la Ecuación 1, se determina que para alimentar correctamente a los seis servomotores se requiere de una fuente de alimentación con una capacidad de corriente igual o superior a 1.5 A. Esta fuente externa se conectará en paralelo a la entrada VCC de cada servomotor para suministrarles la tensión de operación requerida. Los

pinos de tierra de los servomotores se unirán a una tierra común y estos a su vez se conectarán a una tierra de la placa, garantizando así que todos los componentes compartan una misma referencia de voltaje. Para controlar cada servomotor de forma independiente, la Raspberry Pi enviará señales PWM a través de seis pines GPIO configurados especialmente para esta función. Variando el ancho de pulso de estas señales, la Raspberry Pi puede modular la posición y velocidad de cada motor por separado. Las conexiones para este subsistema se detallan en la Figura 18.

**Figura 18**

*Conexiones del subsistema de dispensación*



*Nota. Autoría*

### 3.5.1.2. Programación del Subsistema de Dispensación

Para automatizar la dispensación de medicamentos desde los seis compartimientos, se deben extraer los valores de hora, minuto y AM/PM para cada compartimiento, estos datos provienen de la base de datos vinculada a la aplicación móvil y son almacenados en una estructura denominada diccionario. Cada compartimiento y sus atributos se guardan en forma de pares clave: valor, donde clave representa el nombre del compartimiento y valor es el conjunto de atributos asignados a ese compartimiento, como son hora, minuto, ampm y la

cantidad. Luego, se define un diccionario denominado datoDefecto con valores predeterminados para los atributos de los compartimientos y se extraen la información del diccionario datos utilizando el método get. Parte de la codificación se observa en la Figura 19.

### Figura 19

*Extracción de los datos almacenados en cada compartimiento*

```

48 #Extraccion de compartimientos
49 datoDefecto={'Hora': '', 'Minuto': '', 'ampm': '', 'cant':'' }
50 comp1= datos.get('Compartimiento 1', datoDefecto)
51 comp2= datos.get('Compartimiento 2', datoDefecto)
52 comp3= datos.get('Compartimiento 3', datoDefecto)
53 comp4= datos.get('Compartimiento 4', datoDefecto)
54 comp5= datos.get('Compartimiento 5', datoDefecto)
55 comp6= datos.get('Compartimiento 6', datoDefecto)

```

*Nota.* Autoría

A continuación, se inicializan las variables necesarias para cada compartimiento y sus atributos con cadenas vacías. El propósito es extraer información específica de manera estructurada para luego realizar la manipulación y conversión de los datos.

Luego, se definen las funciones entero, enteroCant y horario. La función entero toma un valor como entrada, elimina cualquier comilla alrededor y verifica si existe algo después de quitar las comillas. Si hay un valor, intenta convertirlo a entero, en caso de tratarse de un valor que convertible a entero. La función enteroCant está diseñada para eliminar las comillas vacías de cantidades. Si el valor está rodeado de comillas vacías o no hay valor, devuelve 0. Si el valor es convertible devuelve el entero, caso contrario devuelve 0. La función horario verifica la información del periodo del día en el que se haya definido el horario. Ya sea am o pm. Si existe el valor, lo devuelve sin alteraciones, en caso de no existir devuelve, noampm. Los valores transformados se asignan a variables correspondientes a cada compartimiento. Además, las cantidades asociadas a cada compartimiento también se convierten a enteros. Finalmente, se crea una lista que almacena las cantidades convertidas de todos los compartimientos. En la Figura 20 se presenta un fragmento de la codificación utilizada para transformar los valores

provenientes de la base de datos a valores enteros y que pueden ser manipulables para ejecutar las acciones del dispensador.

## Figura 20

*Conversión de los valores a enteros*

```
180 | #Compartimiento 1
181 | horaComp1=entero(hComp1)
182 | minutosComp1=entero(mComp1)
183 | ampmComp1= horario(ampComp1)
184 | #Compartimiento 2
185 | horaComp2=entero(hComp2)
186 | minutosComp2=entero(mComp2)
187 | ampmComp2= horario(ampComp2)
```

*Nota.* Autoría

Posteriormente, se utiliza el módulo `datetime` para obtener la hora actual del sistema y se almacena en otra variable denominada `hora_total`. Esta variable se extrae en un formato de 24 horas, se convierte a entero y se almacena en dos variables. Se verifica si la hora actual es mayor que 12, lo que indica que es después del mediodía. De ser así, se resta 12 de la variable definida para la hora actual y de esta forma se representa la hora en el formato de 12 horas. De mismo modo, se extraen los minutos, se convierten a entero y se almacenan en una nueva variable. Para determinar si se trata de am o pm en el nuevo formato de hora, es necesario basarse en el valor de la segunda variable para la hora actual. Si es mayor o igual se establece el horario como pm, caso contrario se establece como am. En la Figura 21 se presenta el fragmento del código implementado para extraer la hora del sistema.

**Figura 21**

*Extraer la hora actual del sistema*

```

231 #Extraer Hora del sistema
232 hora_total= datetime.now().time()
233 horaActualS= int(hora_total.strftime("%H"))
234 horaActualS2= int(hora_total.strftime("%H"))
235 if horaActualS>12:
236     horaActualS= horaActualS-12
237 minutoActualS = int(hora_total.strftime("%M"))
238 horario=' '
239 if horaActualS2>=12:
240     horario='PM'
241 else:
242     horario='AM'
243 print('horario del reloj', horario)
244 print(horaActualS, "hora del reloj**")
245 print(minutoActualS, "minutos del reloj")

```

*Nota. Autoría*

A continuación, se inicializa una variable booleana para controlar si el compartimiento debe ser accionado o no. Y se crea una lista vacía con seis elementos, destinada a registrar el número de compartimiento que ha sido activado. Luego, se verifica si la hora programada para el compartimiento coincide con la hora actual en formato de 12 horas. Si esta condición se cumple, se verifica si los minutos programados coinciden con los minutos actuales. En caso de que se cumplieran estas dos condiciones, se realiza un llamado a la función para notificar al usuario que es la hora de tomar su medicamento. Este proceso se repite para verificar si se cumplen las condiciones de tiempo para activar cada compartimiento individualmente. En la Figura 22, se puede apreciar el fragmento referente al compartimiento 1.

**Figura 22**

*Encontrar coincidencias entre la hora actual y la hora programada*

```

262 #Condiciones Compartimiento 1
263 if horaComp1==horaActualS and horario==ampmComp1:
264     print('Hora cumple')
265     if minutosComp1==minutoActualS:
266         #Ejecuta el bot con la notificación
267         notifRecordatorio('Compartimiento 1')
268         varControl=True
269         lista[0]=1

```

*Nota. Autoría*

Se crea una función que recibe dos argumentos para determinar si se debe activar el compartimiento y cuál es el compartimiento que debe activarse. Si el primer argumento de la función es verdadero, se llama a la función pantalla, cuya función es la de presentar un botón al usuario que le permite desplegar la cámara para el reconocimiento facial. Si el usuario es verificado correctamente, muestra la notificación de retiro exitoso. Y se llama a la función para activar los servos según la lista proporcionada. En caso de que la verificación del usuario no se haya completado, se envía una notificación de retiro no exitoso. En la Figura 23 se aprecia el código implementado para ejecutar las acciones de reconocimiento facial y activar los servomotores.

### Figura 23

*Función para activar servomotores*

```

326 | #Función activadora rec facial y servos
327 | def Accionador(condicionFinal, lista):
328 |     global listaCantidades
329 |     if condicionFinal == True:
330 |         correcto= False
331 |         correcto = pantalla()
332 |         if correcto==True:
333 |             notifRetiroSi()
334 |             print('lista')
335 |             funServos(lista)
336 |             #Mostrar la cantidad
337 |             for i, elemento in enumerate(lista):
338 |                 if isinstance(elemento, int) or isinstance(elemento, float):
339 |                     if listaCantidades[i] > 0:
340 |                         listaCantidades[i] -= 1

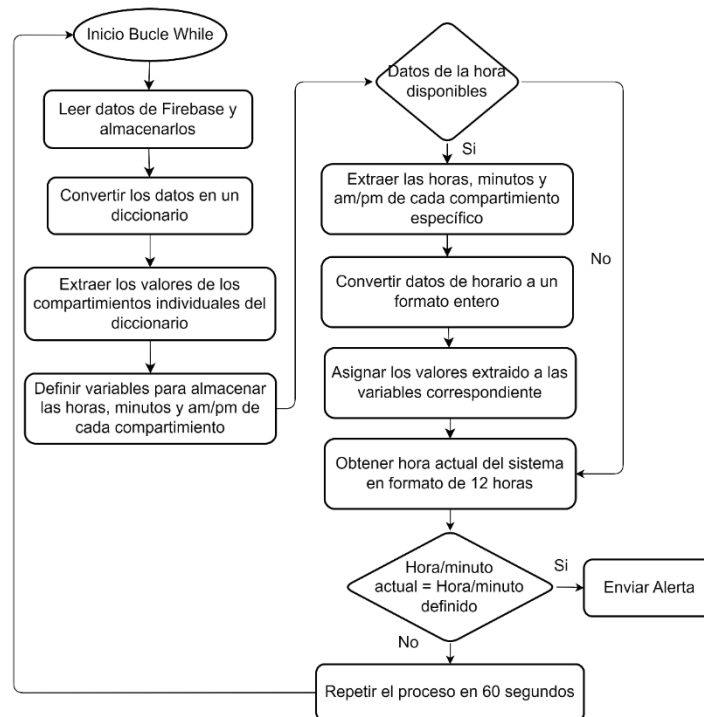
```

*Nota. Autoría*

En la Figura 24 se muestra el flujograma de programación de los compartimientos o contenedores de medicamentos del dispensador.

**Figura 24**

*Diagrama de flujo de programación de los compartimientos*



*Nota. Autoría*

Para controlar la velocidad del servomotor usando la modulación de ancho de pulso se utiliza la librería RPi.GPIO en Python, que permite controlar los pines GPIO de la Raspberry Pi. Una vez definido este parámetro, se crea una función que recibe como argumento el número del pin GPIO al que se encuentra conectado el servomotor. Después, se configura este pin como una salida, es decir, que se envíe una señal eléctrica desde la Raspberry hacia el servomotor para que este puede ejecutar acciones. Posteriormente, se definen dos objetos, el primero se utiliza para controlar la posición del servomotor y recibe como parámetros al número de pin y la frecuencia de la señal PWM. El segundo objeto PWM define un ciclo de trabajo inicial en 0, durante este ciclo el servo mantiene una posición neutral. A través de la función `set_speed` que toma como parámetro la velocidad del servomotor, se cambia el ciclo de trabajo de la señal

en función de la velocidad de entrada y luego se usa el método `ChangeDutyCycle` para aplicar este nuevo ciclo de trabajo a la señal PWM, como se observa en la Figura 25.

## Figura 25

### Programación de lo servomotores

```

8      GPIO.setmode(GPIO.BCM)
9      GPIO.setup(servo_pin, GPIO.OUT)
10
11     pwm = GPIO.PWM(servo_pin, 50) # Frecuencia de PWM: 50 Hz
12     pwm.start(0) # Iniciar el PWM con un ciclo de trabajo del 0%
13
14     def set_speed(speed):
15         duty_cycle = speed * 0.1 + 1.7 # Calcular el ciclo de trabajo necesario para la
16         pwm.ChangeDutyCycle(duty_cycle)
17         print(duty_cycle)
18
19     try:
20         set_speed(1) # Establecer la velocidad máxima en sentido horario (valor de 10)
21         time.sleep(0.148) # Mantener la velocidad máxima durante 5 segundos
22         set_speed(0) # Detener el servo

```

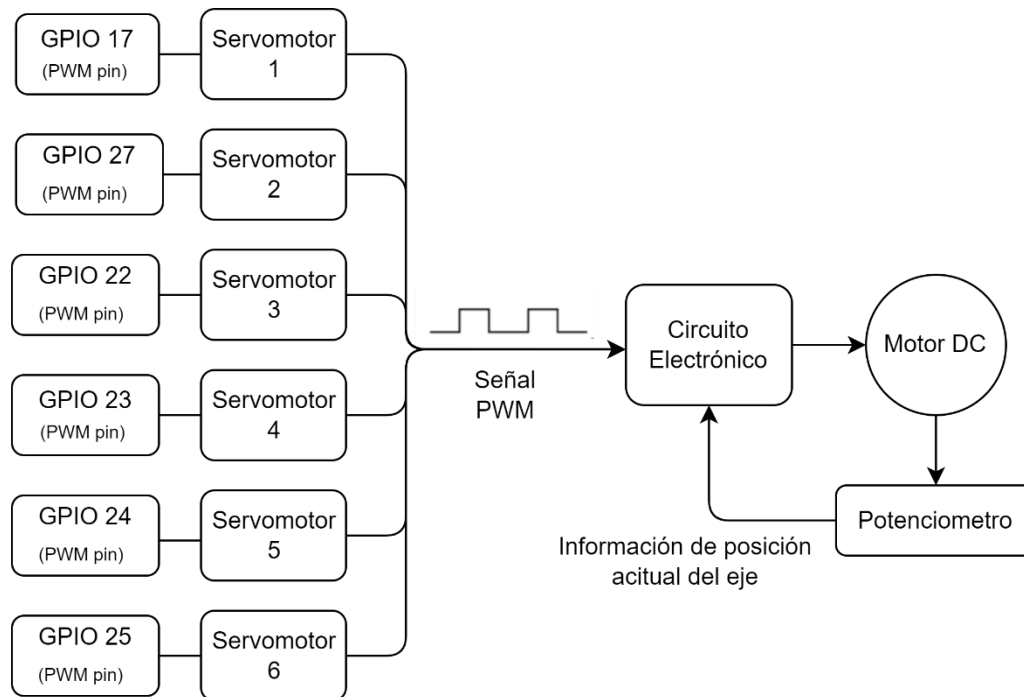
### Nota. Autoría

Las llamadas a `set_speed(1)` y `set_speed(0)` establecen la velocidad del servo motor en 1 y 0, respectivamente. Se introduce una pausa mediante la función `time.sleep` después de cada cambio de velocidad, lo que permite que el servomotor se desplace a la posición correspondiente. En la Figura 26, se muestra un diagrama de funcionamiento del servomotor conectado a los pines correspondientes en la Raspberry Pi.



**Figura 26**

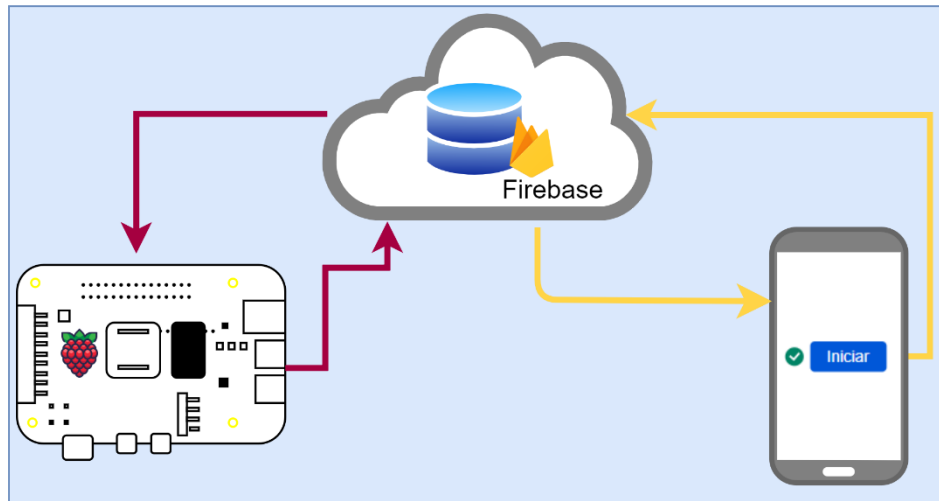
*Diagrama de bloques del funcionamiento de los servomotores*



*Nota. Autoría*

### **3.5.2. Almacenamiento en la Nube**

Se ha optado por utilizar los servicios de Firebase para almacenar y sincronizar los datos en la nube, de esta forma los registros de pastillas y su horario de dispensación se guardarán de forma segura. Gracias a su característica de sincronización en tiempo real, los cambios que se realicen en la aplicación se reflejarán rápidamente en la Raspberry para que pueda ejecutar las diferentes acciones. En la Figura 27, se puede apreciar la arquitectura para la conexión entre la aplicación, Firebase y la Raspberry Pi.

**Figura 27***Almacenamiento en la nube**Nota. Autoría*

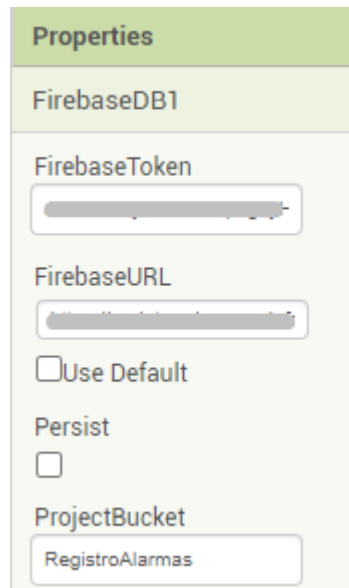
### 3.5.2.1. Conexión Raspberry Pi – Firebase

Para realizar la conexión entre el script de Python que correrá en la Raspberry Pi y Firebase, es necesario tener el proyecto ya creado en Firebase y la base de datos para poder interactuar. Por otro lado, en la Raspberry se debe instalar la biblioteca Pyrebase de Python para poder interactuar con los servicios de Firebase desde la computadora de placa única. Luego, se definió un diccionario que contiene los datos de configuración necesarios para establecer la conexión, como lo son la clave de API y el dominio de autenticación del proyecto. Además, la URL de la base de datos y el bucket de almacenamiento de Firebase. Esta configuración se utiliza para inicializar la biblioteca Pyrebase. Finalmente, se obtiene una referencia con la que se pueden acceder y manipular los datos disponibles en la base de datos desde el código de Python. En Figura 28 se puede observar el código implementado en un script de Python.



**Figura 29**

*Configuración de la conexión de la aplicación móvil – Firebase*



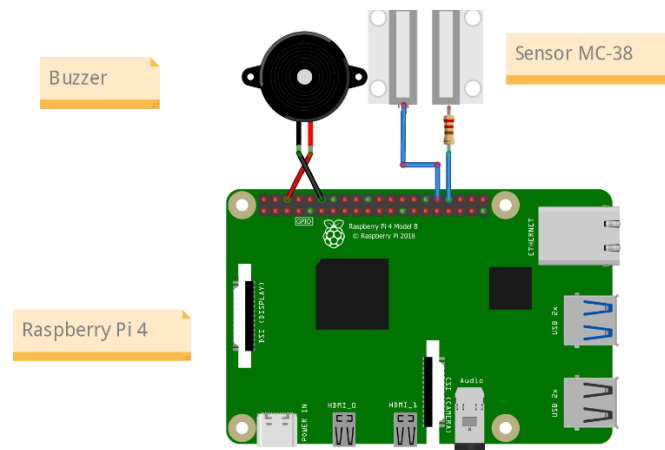
*Nota.* Autoría

### **3.5.3. Subsistema de Seguridad**

Dentro del subsistema de seguridad se encarga de monitorizar el estado del dispositivo, en específico la compuerta para el ingreso de pastillas a los compartimientos, dónde emitirá una alarma al momento de reconocer que la compuerta se encuentra en estado abierto, siendo este el caso emitirá un sonido que alarme al usuario donde indicará que la compuerta se encuentra abierta, caso contrario la alarma queda inactiva y no emite ningún sonido.

#### **3.5.3.1. Diagrama de conexiones del subsistema de Seguridad**

Para el subsistema sea requerido de un sensor MC-38 el cual fue seleccionado debido a sus características, el cual trabaja enviando señales de unos y ceros, y la Raspberry leer a los datos enviados por el sensor y si el sensor envía un uno la Raspberry lo reconoce y mandará la señal hacia el buzzer que está conectado directamente a la Raspberry. El diagrama de conexión de este subsistema se ilustra en la Figura 30.

**Figura 30***Diagrama de Subsistema de Seguridad**Nota: Autoría*

### 3.5.3.2. Programación del Subsistema de Seguridad

Para garantizar la seguridad de los medicamentos, es necesario mantener el ambiente en el que se encuentran en condiciones estériles. Para ello, se ha diseñado un sistema de alarma que se activa cuando la compuerta que da acceso a los compartimientos de los medicamentos está abierta. Si la compuerta está cerrada, el sistema no emite ninguna alarma. De esta manera, se evita que cualquier impureza pueda contaminar los medicamentos y causar su deterioro.

En resumen, el sistema de alarma funciona de la siguiente manera:

- Si la compuerta está abierta: El sistema emite una alarma para alertar a los usuarios de que los compartimientos de los medicamentos están expuestos a la contaminación.
- Si la compuerta está cerrada: El sistema no emite ninguna alarma, ya que los compartimientos están protegidos de la contaminación.

Para esto, es necesario leer el estado del sensor, configurar condiciones iniciales y en el momento en el que se cumplan dichas condiciones, se activa o desactiva el sensor, como se observa en la Figura 31.

**Figura 31**

*Programación del subsistema de seguridad*

```

1  import RPi.GPIO as GPIO
2  import time
3
4  # Pin donde está conectado el sensor MC38
5  sensorPin = 12
6  buzzerPin = 18
7
8  GPIO.setmode(GPIO.BCM)
9
10 # Sensor pin como entrada
11 GPIO.setup(sensorPin, GPIO.IN)
12 # Sensor pin como salida
13 GPIO.setup(buzzerPin, GPIO.OUT)
14
15
16 try:
17     while True:
18         # Leer el estado del sensor
19         sensorState = GPIO.input(sensorPin)
20
21         # Verificar si se detectó un contacto
22         if sensorState == GPIO.HIGH:
23             print("Contacto detectado: ABIERTO")
24             # Encender el buzzer
25             GPIO.output(buzzerPin, GPIO.HIGH)
26             time.sleep(0.2)
27             # Apagar el buzzer
28             GPIO.output(buzzerPin, GPIO.LOW)
29             time.sleep(0.2)
30         else:
31             print("Contacto detectado: CERRADO")
32             # Apagar el buzzer
33             GPIO.output(buzzerPin, GPIO.LOW)

```

*Nota. Autoría*

### 3.5.4. Subsistema de Reconocimiento

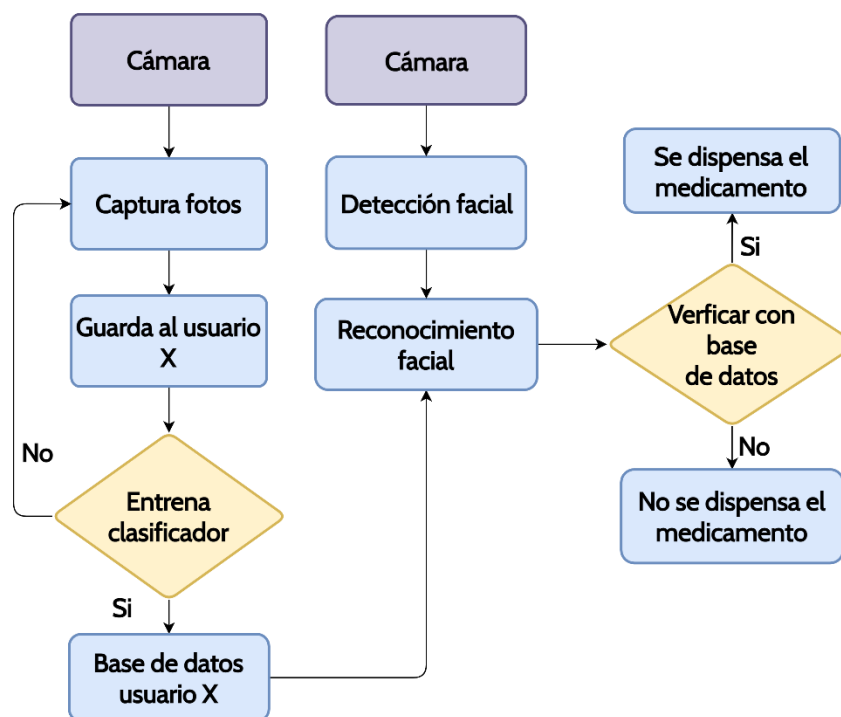
El subsistema de reconocimiento se enfoca en la identificación del usuario, utilizando técnicas como el reconocimiento facial para verificar su identidad antes de dispensar los medicamentos. Esto asegura que solo el usuario autorizado tenga acceso a los medicamentos, evitando errores o confusiones. Por otro lado, con esta funcionalidad se pretende evitar que se dispensen dosis duplicadas. Si el sistema no reconoce al usuario, no se dará paso a la dispensación, previniendo la ingesta indebida del medicamento.

El proceso comienza con la captura de la imagen facial del usuario a través de la cámara. Una vez capturada la imagen, se guarda en la base de datos asociada a un usuario denominado usuario X. Después de guardar la imagen, se procede a entrenar un clasificador, que es un algoritmo que aprende a reconocer las características faciales específicas del usuario a partir de las fotos almacenadas en la base de datos. Una vez que el clasificador está entrenado, se

crea una base de datos para el usuario X. Esta base de datos contendrá las características faciales y cualquier otra información relevante. Luego, se procede al reconocimiento facial, en donde el sistema realiza una comparación de las características faciales de una nueva foto capturada con las características almacenadas en la base de datos del usuario X. Si existe coincidencia, se procede a dispensar la pastilla, caso contrario el sistema no se activará. Este proceso se representa la Figura 32.

**Figura 32**

*Diagrama general del proceso de reconocimiento facial*



*Nota.* Elaborado por el autor

### 3.5.4.1. Conexiones del Subsistema de Reconocimiento Facial

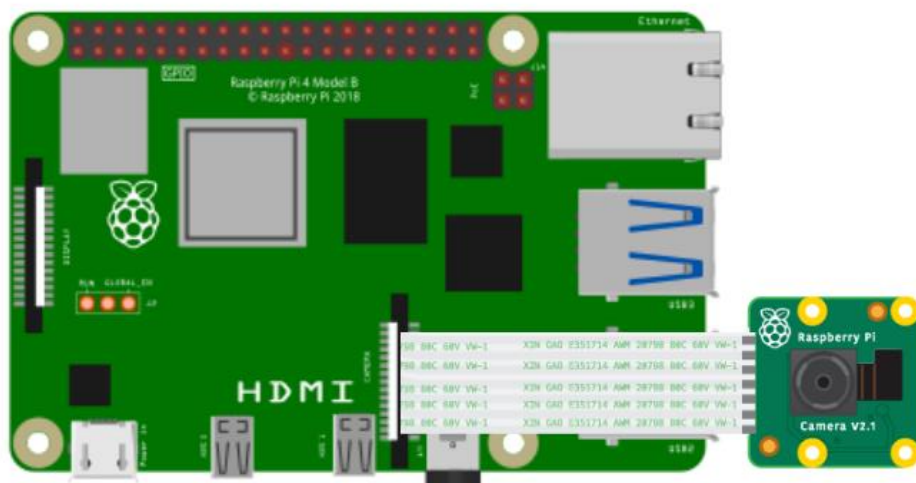
La Raspberry Pi dispone de un conector exclusivo para la conexión de cámaras denominado puerto CSI (Camera Serial Interface). Este puerto se utiliza para la transmisión de imágenes desde la cámara hacia la Raspberry. La estructura del cable CSI de la cámara consiste en una cinta plana de 15 pines, protegidos por una capa aislante para evitar interferencias. Tiene

una longitud estándar de aproximadamente 15 cm y está equipado con un conector en forma de Z en el extremo que se conecta a la placa, esta característica asegura una conexión estable, previniendo que se desconecte accidentalmente.

En la Raspberry Pi, el puerto CSI está ubicado entre la salida de audio/video y uno de los puertos micro HDMI, para asegurar que la conexión se realice de forma exitosa y brindar seguridad, es necesario utilizar la pestaña de retención en el conector. La Figura 33 muestra el diagrama de conexión de este subsistema.

### Figura 33

*Conexiones del subsistema de reconocimiento*



*Nota.* Autoría

La Figura 34 es una representación visual que ilustra cómo se conecta y comunica la pantalla táctil con la Raspberry Pi. En este esquema, se utiliza el puerto DSI (Display Serial Interface), un conector de 15 pines que permite una transmisión de datos de video desde la Raspberry Pi hacia la pantalla, transmitiendo los datos uno tras otro en una secuencia óptima para garantizar una experiencia visual fluida y nítida. Para la configuración, se requiere conectar físicamente el conector DSI de la pantalla al puerto DSI de la Raspberry.



## Figura 34

### Conexión pantalla – Raspberry Pi



#### Nota. Autoría

La Raspberry Pi, en combinación con la cámara, son los dos componentes de hardware clave para el desarrollo de esta sección. A través de la Raspberry Pi, es posible implementar y ejecutar el algoritmo de reconocimiento facial.

Para empezar, es necesario configurar el entorno, es decir, instalar un sistema operativo en la Raspberry Pi, en este caso se utilizará Raspbian. La imagen de este sistema operativo está disponible en el sitio oficial de Raspberry Pi. Una vez que se haya descargado, se procede a grabar la imagen en la tarjeta microSD. Al finalizar este proceso, la tarjeta esta lista para ser insertada en la ranura correspondiente de la Raspberry Pi. Por otro lado, es necesario tener habilitada la conexión a la Raspberry Pi Camera Module.

#### 3.5.4.2. Datos de Entrenamiento

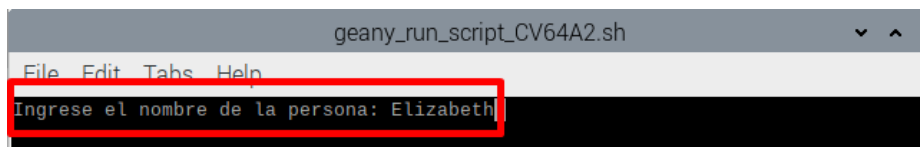
Como parte del proceso es necesario asegurarse de que el lenguaje de programación Python esté disponible (generalmente ya está instalado en las Raspberry Pi), ya que este será el lenguaje que se utilizará para implementar reconocimiento facial. Otra de las dependencias necesarias es la biblioteca de OpenCV, con ella se puede tener acceso a funciones para el

procesamiento de imágenes y el reconocimiento facial. También se debe incluir la librería que permite la interacción con el sistema operativo y los componentes específicos de la cámara.

La programación inicia definiendo el nombre de la persona cuyo rostro se va a capturar, como muestra la Figura 35, una vez que se define el nombre, se crea una carpeta que almacenará las imágenes capturadas del usuario en la carpeta images. Esto lo hace a través de un bucle continuo que captura cien imágenes, una vez que sobrepasa este valor, el bucle se rompe.

### Figura 35

*Agregar nuevo usuario a la base de datos de imágenes*



*Nota.* Autoría

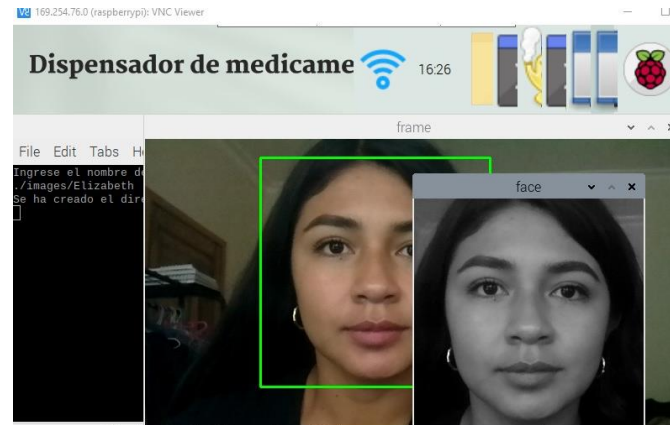
Luego, se lleva a cabo la detección de facial, este paso permite localizar, delimitar e identificar la región de la imagen en donde se encuentra un rostro. OpenCV proporciona un clasificador de cascada pre entrenado, denominado Haar, el cual ha demostrado ser efectivo en la detección de rostros frontales. El algoritmo está dado en el formato XML, y se lo puede obtener directamente de la página oficial de OpenCV. En este archivo se encuentran contenidos los parámetros y características necesarios para detectar rostros.

A través de un bucle, se obtiene un fotograma individual o una imagen estática del video que se está capturando desde la cámara. Posteriormente, se realiza una estandarización del tamaño de la imagen, al tener un tamaño definido de las imágenes, se reduce la carga computacional y se simplifica el manejo de cada una de ellas. Luego de detectar los rostros en la imagen en escala de grises, se dibuja un rectángulo en la imagen original y se procede a recortar esa área, es decir, extraer únicamente aquella región donde se encuentra un rostro como

se observa en la Figura 36 y se guarda en la carpeta images. Por otro lado, el contador se incrementa para llevar un registro del número de imágenes capturadas.

### Figura 36

#### *Detección del rostro*

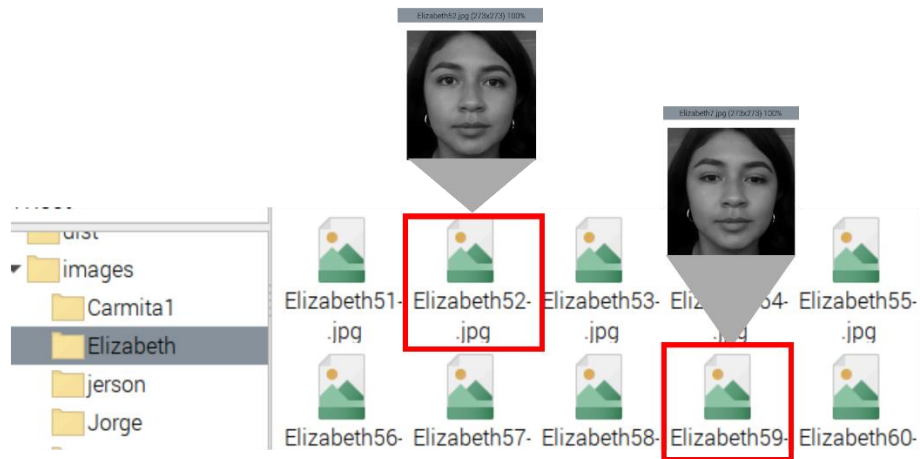


#### *Nota. Autoría*

En la Figura 37, se puede observar todas las imágenes de los rostros redimensionadas, convertidas a escala de grises y guardadas en la carpeta especificada. Estas imágenes serán la base para que el modelo aprenda a distinguir las características únicas de cada usuario. Este proceso de preprocesamiento es muy importante para preparar los datos antes de entrenar el modelo de reconocimiento facial, puesto que se garantiza que el modelo reciba datos consistentes durante el entrenamiento.

**Figura 37**

*Conjunto de imágenes del usuario*

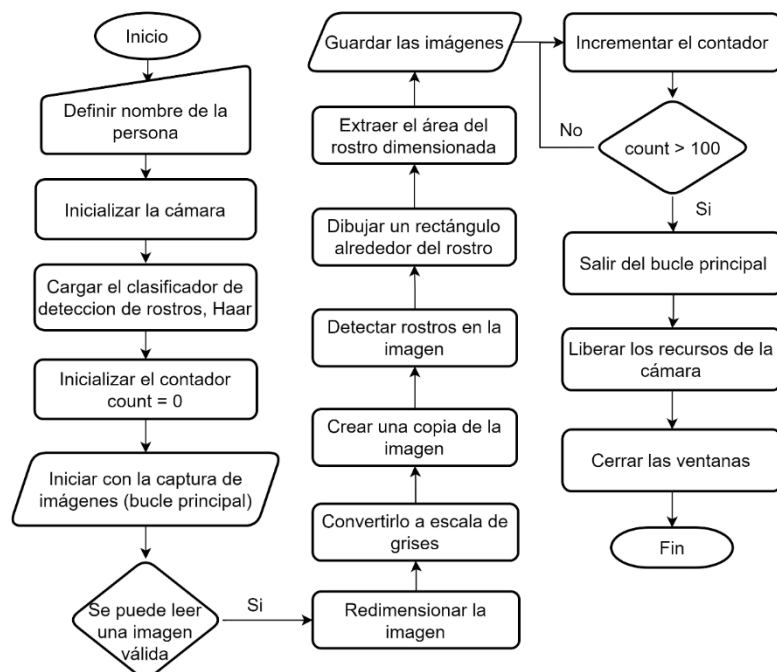


*Nota. Autoría*

Todo el proceso de programación del código para obtener los datos para el entrenamiento se encuentra detallado en flujograma de la Figura 38, que describe paso a paso las acciones necesarias para adquirir las imágenes que contienen los rostros a reconocer.

**Figura 38**

*Flujograma de la programación para obtener los datos de entrenamiento*



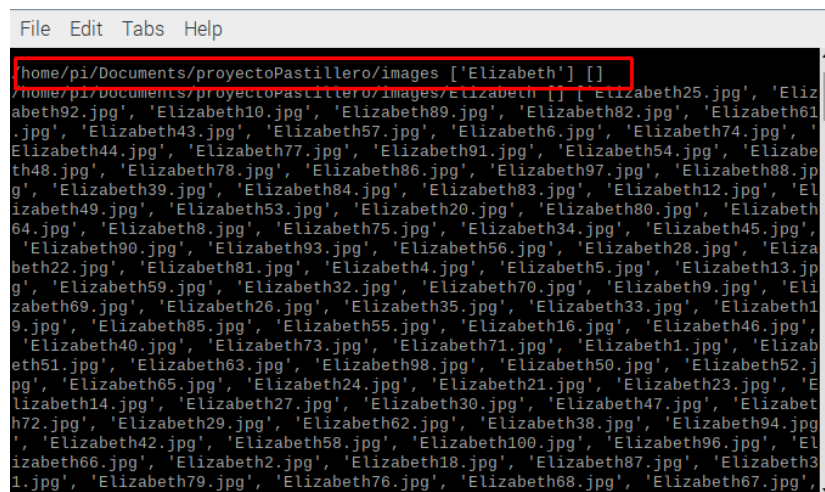
*Nota. Autoría*

### 3.5.4.3. Entrenar el Modelo

Las imágenes de los rostros ya guardadas se utilizan como datos de entrada para el entrenamiento del algoritmo de reconocimiento. Para empezar, es necesario establecer la ruta de la carpeta que contiene las imágenes de los rostros. A continuación, se obtiene una lista de personas presentes en la carpeta y se inicializan las listas para el almacenamiento de las etiquetas y los datos de los rostros de forma temporal. A medida que se recorre cada persona en la lista, se procesan las imágenes de los rostros que correspondan a esa persona, una vez procesadas, se les asigna un número único y se almacena en la lista de etiquetas. Luego, se crea un objeto de reconocimiento facial utilizando el método LBPH, y se entrena el modelo utilizando las imágenes de los rostros y sus respectivas etiquetas, este proceso se puede observar en la Figura 39.

**Figura 39**

*Procesamiento de las imágenes correspondiente al usuario de prueba*



```
File Edit Tabs Help
/home/pi/Documents/proyectoPastillero/images ['Elizabeth'] []
/home/pi/Documents/proyectoPastillero/images/Elizabeth [] ['Elizabeth25.jpg', 'Elizabeth92.jpg', 'Elizabeth10.jpg', 'Elizabeth89.jpg', 'Elizabeth82.jpg', 'Elizabeth61.jpg', 'Elizabeth43.jpg', 'Elizabeth57.jpg', 'Elizabeth6.jpg', 'Elizabeth74.jpg', 'Elizabeth44.jpg', 'Elizabeth77.jpg', 'Elizabeth91.jpg', 'Elizabeth54.jpg', 'Elizabeth48.jpg', 'Elizabeth78.jpg', 'Elizabeth86.jpg', 'Elizabeth97.jpg', 'Elizabeth88.jpg', 'Elizabeth39.jpg', 'Elizabeth84.jpg', 'Elizabeth83.jpg', 'Elizabeth12.jpg', 'Elizabeth49.jpg', 'Elizabeth53.jpg', 'Elizabeth20.jpg', 'Elizabeth80.jpg', 'Elizabeth64.jpg', 'Elizabeth8.jpg', 'Elizabeth75.jpg', 'Elizabeth34.jpg', 'Elizabeth45.jpg', 'Elizabeth90.jpg', 'Elizabeth93.jpg', 'Elizabeth56.jpg', 'Elizabeth28.jpg', 'Elizabeth22.jpg', 'Elizabeth81.jpg', 'Elizabeth4.jpg', 'Elizabeth5.jpg', 'Elizabeth13.jpg', 'Elizabeth59.jpg', 'Elizabeth32.jpg', 'Elizabeth70.jpg', 'Elizabeth9.jpg', 'Elizabeth69.jpg', 'Elizabeth26.jpg', 'Elizabeth35.jpg', 'Elizabeth33.jpg', 'Elizabeth19.jpg', 'Elizabeth85.jpg', 'Elizabeth55.jpg', 'Elizabeth16.jpg', 'Elizabeth46.jpg', 'Elizabeth40.jpg', 'Elizabeth73.jpg', 'Elizabeth71.jpg', 'Elizabeth1.jpg', 'Elizabeth51.jpg', 'Elizabeth63.jpg', 'Elizabeth98.jpg', 'Elizabeth50.jpg', 'Elizabeth52.jpg', 'Elizabeth65.jpg', 'Elizabeth24.jpg', 'Elizabeth21.jpg', 'Elizabeth23.jpg', 'Elizabeth14.jpg', 'Elizabeth27.jpg', 'Elizabeth30.jpg', 'Elizabeth47.jpg', 'Elizabeth72.jpg', 'Elizabeth29.jpg', 'Elizabeth62.jpg', 'Elizabeth38.jpg', 'Elizabeth94.jpg', 'Elizabeth42.jpg', 'Elizabeth58.jpg', 'Elizabeth100.jpg', 'Elizabeth96.jpg', 'Elizabeth66.jpg', 'Elizabeth2.jpg', 'Elizabeth18.jpg', 'Elizabeth87.jpg', 'Elizabeth31.jpg', 'Elizabeth79.jpg', 'Elizabeth76.jpg', 'Elizabeth68.jpg', 'Elizabeth67.jpg',
```

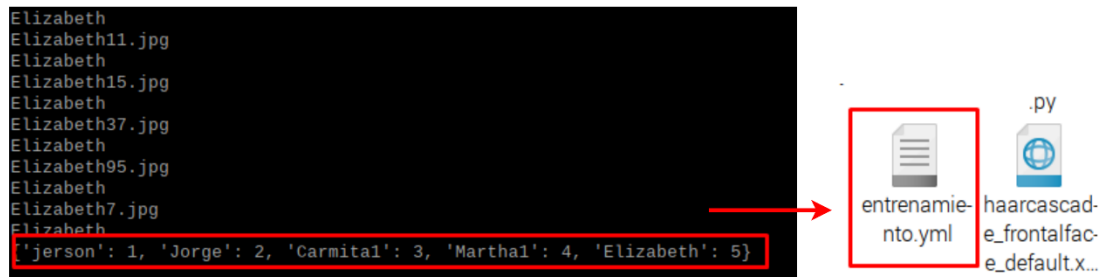
*Nota. Autoría*

Al finalizar, se guarda el modelo entrenado en un archivo “yaml” para su uso posterior en el reconocimiento del usuario del dispensador de medicamentos, como se observa en Figura 40. Es decir, el usuario se acerca al dispensador y sistema de reconocimiento se activará, la

cámara captura una imagen del rostro del usuario y esta será procesada, de manera que se extraen las características más importantes y ser comparan con las características almacenadas en la base de datos. En caso de encontrar coincidencia entre las dos características, se considera al usuario como autorizado.

## Figura 40

### Entrenamiento del modelo

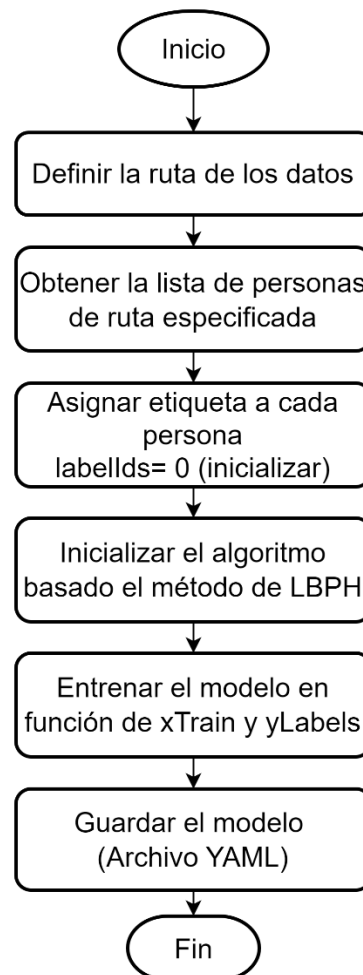


### Nota. Autoría

El diagrama de flujo de la Figura 41, representa el proceso de recopilación, procesamiento y entrenamiento de imágenes de rostros para efectuar el proceso de reconocimiento facial. En primer lugar, se detectan las caras en las imágenes, se les asigna etiquetas únicas a cada persona, se crea un modelo de reconocimiento basado en LBPH y se entrena con las imágenes procesadas.

**Figura 41**

*Diagrama de flujo del algoritmo de entrenamiento*



*Nota.* Autoría

#### **3.5.4.4. Algoritmo Implementado en la Interfaz del Dispensador**

El algoritmo que se ha implementado en la interfaz del dispensador utiliza el modelo previamente entrenado. Detecta rostros en cada fotograma capturado utilizando el clasificador de cascada frontal Haar. Luego, se realiza el reconocimiento facial mediante un modelo previamente entrenado y se obtiene un identificador de la persona reconocida junto con un valor de confianza.

En cuanto a la codificación, configura un sistema de reconocimiento facial en una Raspberry Pi. El sistema carga un diccionario de etiquetas de personas desde un archivo y crea

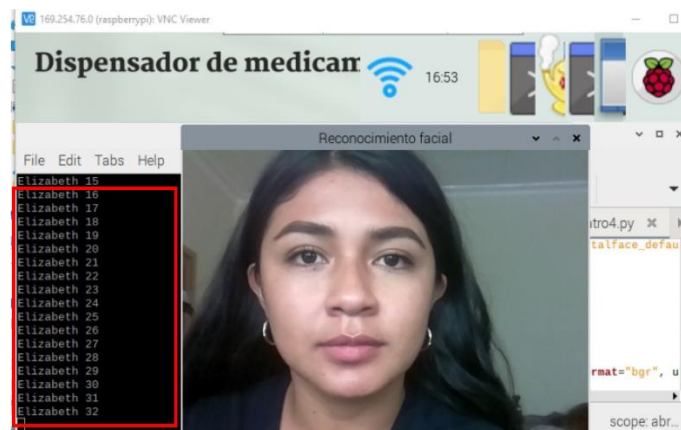
un hilo para controlar un relé. La cámara de la Raspberry Pi se configura para capturar imágenes en resolución 480x480 y 30 cuadros por segundo.

En un bucle principal, el sistema captura imágenes continuamente desde la cámara. Para cada imagen, el sistema la convierte a escala de grises y detecta caras utilizando el clasificador Haar. La imagen se muestra en una ventana llamada "Reconocimiento facial" y las caras detectadas se analizan.

El sistema utiliza un modelo previamente entrenado basado en LBPH para intentar predecir la identidad de cada rostro. La confianza de la predicción se compara con un umbral. Si la confianza es suficiente, el sistema inicia un hilo para activar el relé. El sistema continúa capturando imágenes hasta que se detectan 50 rostros, como se muestra en la Figura 42. En ese momento, el sistema cierra la cámara y destruye las ventanas, finalizando la ejecución del programa.

### Figura 42

#### *Algoritmo de reconocimiento en ejecución*



*Nota.* Autoría

### 3.5.5. Programación de Alertas

Uno de los criterios más importantes en el diseño del dispensador, es la generación de alertas que faciliten la toma de medicamentos al usuario. A través de estas alertas el usuario



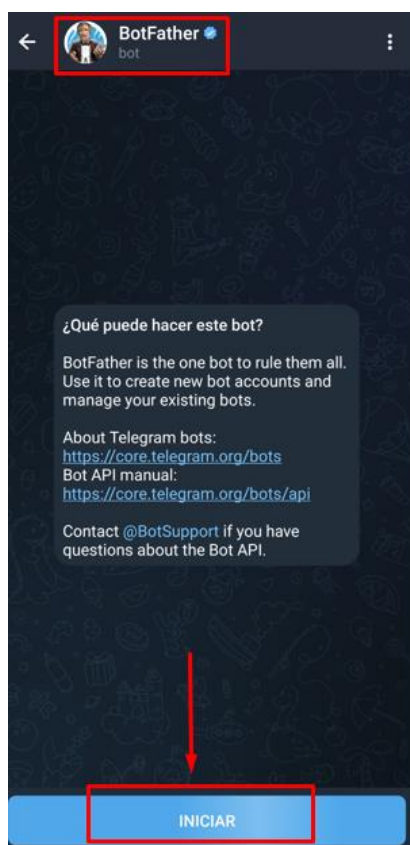
puede recordar cuando es el momento exacto para la toma de su medicación, en especial para aquellos que los consumen regularmente y en horarios específicos.

Una forma innovadora para el envío de alertas es a través de bots de Telegram, que permiten enviar recordatorios personalizados y oportunos directamente a los usuarios a través de esta plataforma. Para realizar esta acción se utiliza BotFather, que es un bot oficial de Telegram para la creación de bots.

Para empezar, en la barra de búsqueda de Telegram, buscar BotFather, posteriormente se despliega una ventana para iniciar la conversación, como se muestra en la Figura 43.

### Figura 43

*Iniciar conversación con BotFather*



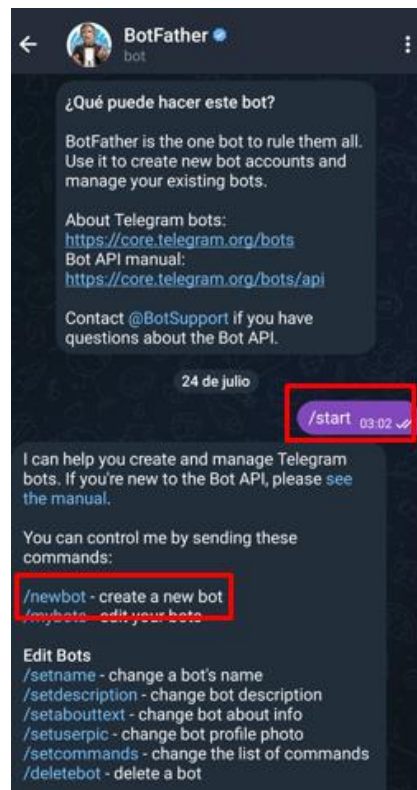
*Nota.* Autoría

Al iniciar una conversación con BotFather, se muestran una serie de comandos para interactuar con él. Estos comandos se dividen en dos categorías: comandos de información y

comandos de administración. Los comandos de información proporcionan información sobre BotFather y sus funciones. Los comandos de administración se utilizan para crear, editar y eliminar bots. Para crear un nuevo bot, se debe enviar el comando /newbot como se observa en la Figura 44.

#### Figura 44

##### *Creación del nuevo bot*



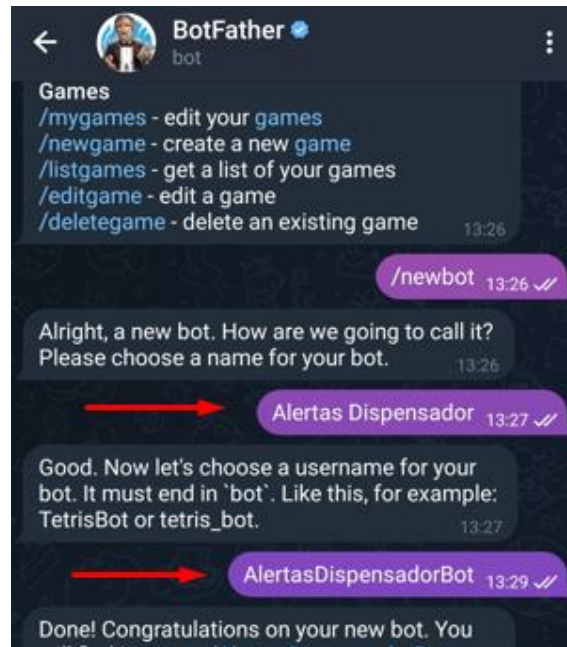
*Nota.* Autoría

Posteriormente, es necesario designar un nombre al bot y también un nombre de usuario para el bot. El nombre de usuario debe ser único y terminar con la palabra "bot". Como muestra la Figura 45. Una vez que se haya elegido el nombre de usuario, BotFather generará un token de acceso único para el bot. Los bots de Telegram tienen dos nombres: uno para que los usuarios los identifiquen y otro para que los distinga de otros bots. El nombre de usuario es el más importante, ya que debe ser único para que los usuarios puedan encontrarlo fácilmente. El token de acceso es una clave que permite a los bots interactuar con la API de Telegram. Es

importante mantener el token de acceso seguro y confidencial para evitar que terceros puedan acceder a él.

### Figura 45

*Nombre y nombre de usuario para el bot.*



*Nota. Autoría*

Para configurar el bot en el teléfono celular, se debe crear un nuevo grupo o canal en la aplicación de mensajería y agregar el bot como miembro. Las notificaciones del bot llegarán al grupo o canal que se haya creado. Se pueden añadir otros usuarios al grupo o canal, como familiares, encargados del cuidado, entre otros.

El código utiliza la API de Telegram para enviar notificaciones a un canal o chat específico. Estas notificaciones se pueden enviar en función de las condiciones que se cumplan dentro del programa principal. El código define tres funciones para enviar notificaciones:

- La función `notifRecordatorio` envía un mensaje de recordatorio para tomar una medicina específica. El parámetro de esta función es el nombre del compartimiento que contiene la pastilla. El mensaje que se envía indica que es hora de tomar la medicina del compartimiento especificado.

- La función `notifRetiroSi` envía un mensaje indicando que el medicamento ha sido retirado exitosamente.
- La función `notifRetiroNo` envía un mensaje indicando que el medicamento no ha sido retirado y se sugiere hacerlo a través de la aplicación.

Estas funciones utilizan la biblioteca `requests` para realizar una solicitud POST a la API de Telegram. La solicitud incluye una URL de la API de Telegram, el token de acceso y el `chat_id` del canal o grupo al que se enviará el mensaje. Parte de la codificación se observa en la Figura 46.

**Figura 46**

*Programación de notificaciones*

```

1 import requests
2
3
4 def notifRecordatorio(valor):
5     text = "Es hora de tomar tu medicina del " + valor + ", acércate a tu dispensador."
6     requests.post('https://api.telegram.org/bot6613723005:AAFFRdTextJFo4kH6A-n29yAWRonA
7     data = {'chat_id': '1670575426', 'text': text})
8 def notifRetiroSi():
9     text = "El medicamento ha sido retirado."
10    requests.post('https://api.telegram.org/bot6613723005:AAFFRdTextJFo4kH6A-n29yAWRonA
11    data = {'chat_id': '1670575426', 'text': text})
12
13 def notifRetiroNo():
14    text = "El medicamento no ha sido retirado. Para hacerlo puedes retirarlo mediante
15    requests.post('https://api.telegram.org/bot6613723005:AAFFRdTextJFo4kH6A-n29yAWRonA
16    data = {'chat_id': '1670575426', 'text': text})

```

*Nota. Autoría*

Cada una de las funciones definidas para el archivo `notificador.py` son invocadas por el código principal implementado en el dispensador en la función denominada `accionador` cuando se cumplen determinadas condiciones:

- `notifRetiroSi()`: Si la condición `condicionFinal` es verdadera y se determina que todo está en orden (`correcto == True`), la función `Accionador` llama a `notifRetiroSi()`
- `notifRetiroNo()`: Si la condición `condicionFinal` es verdadera, pero se detecta algún problema (`correcto == False`), la función `Accionador` llama a `notifRetiroNo()`.

### 3.5.6. Programación de la Interfaz del Dispensador

La pantalla táctil del dispensador permite al usuario controlar el dispositivo, incluyendo la conexión a la red wifi, la configuración de la hora, el estado de los compartimientos y el reconocimiento facial. También se desarrolló una aplicación de escritorio con Tkinter que muestra información sobre el estado de los compartimientos. La función actualizar\_labels recupera datos de Firebase cada 60 segundos y los actualiza en la interfaz de usuario. En la Figura 47 se ilustra la interfaz.

**Figura 47**

*Interfaz para mostrar el estado de los compartimientos*



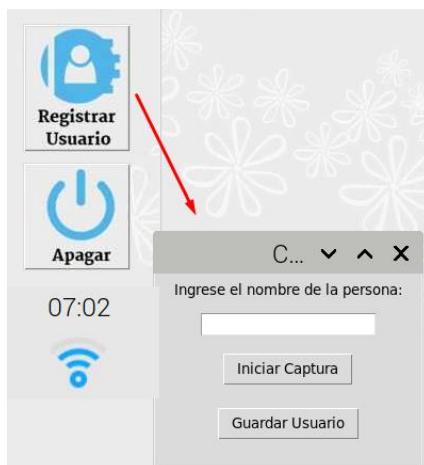
Compartimiento	Medicamento	Cantidad
1	paracetamol	2
2	Ibuprofeno	4
3	Losartan	4
4		0
5		0
6	Predsidona	6

*Nota. Autoría*

Al presionar el botón "Registrar usuario", se abre una ventana emergente, como se ilustra en la Figura 48 que permite al usuario registrar su rostro. Para ello, debe ingresar su nombre y luego presionar el botón "Iniciar captura". Esto abrirá la cámara y el usuario deberá tomarse las muestras faciales requeridas. Una vez que se hayan tomado las muestras, el botón "Guardar usuario" guardará al nuevo usuario en el sistema.

## Figura 48

*Acciones del botón registrar usuario*



*Nota. Autoría*

El botón "Apagar" abre una ventana emergente que permite al usuario apagar o reiniciar el dispensador. El ícono de "Wireless" permite conectarse a Internet, lo cual es necesario para ejecutar las diferentes acciones del dispensador.

### **3.5.7. Diseño de Aplicación Android**

Uno de los objetivos planteados para el desarrollo de este trabajo, es la implementación de una aplicación móvil específica para el control del dispensador. Con esta herramienta, el usuario puede realizar una gestión optimizada de la medicación disponible en cada uno de los contenedores. En otras palabras, a través de la aplicación, el usuario podrá programar el horario, dosis y recordatorios personalizados para cada medicamento.

Para su desarrollo se hará uso de la plataforma App Inventor. Al tratarse de una aplicación dirigida a adultos mayores, es necesario tomar en cuenta algunas consideraciones en cuanto al diseño, como el uso de colores suaves y claros que contrasten entre sí, fuentes grandes y fáciles de leer, íconos simples y elementos bien espaciados.

Entre los colores óptimos para la aplicación resaltan el azul, pues se considera que es un color calmante y relajante que puede ayudar a reducir el estrés, al igual que el verde. A estos

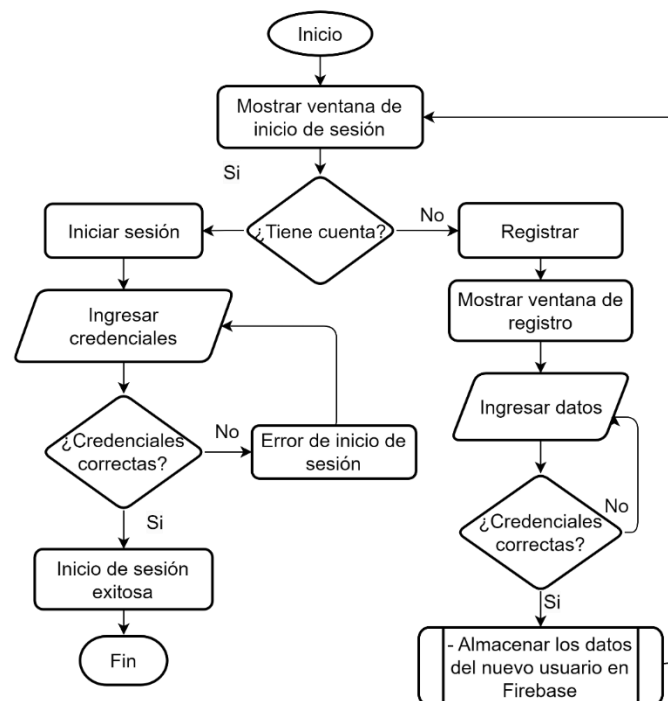
se suma el amarillo que se considera un color que puede ayudar a mejorar el estado de ánimo y es fácil de ver y finalmente el blanco que brinda una sensación de espacio y claridad.

### 3.5.7.1. Inicio de Sesión y Registro

La primera pantalla está dirigida al inicio de sesión y registro de nuevos usuarios. Esta ventana proporciona a los usuarios la posibilidad de ingresar las credenciales existentes o crear una nueva cuenta. Una vez que se haya registrado, la información será guardada en la base de datos. Al ingresar con las credenciales correctas, se desplegará la ventana principal, en donde se encuentran las funcionalidades y características completas de la aplicación. En la Figura 49 se puede apreciar el diagrama de flujo de este proceso.

**Figura 49**

*Diagrama de flujo de la ventana de inicio y registro*



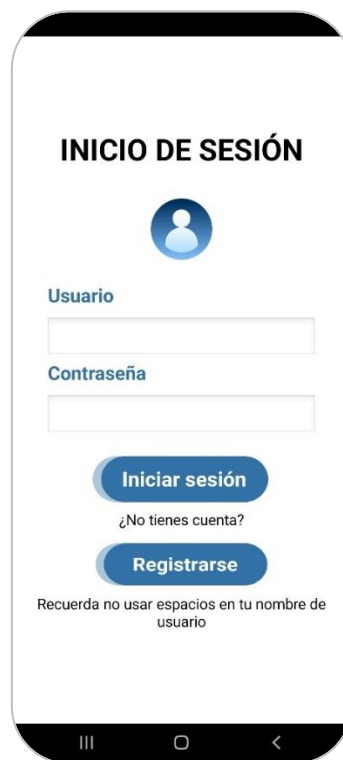
*Nota.* Autoría

La ventana de inicio de sesión es la pantalla que aparece cuando el usuario abre la aplicación móvil por primera vez. Por lo que el diseño ha sido pensado para ofrecer una interfaz amigable y funcional, donde los usuarios puedan ingresar sus credenciales de manera confiable.


La ventana incluye los campos necesarios para que el usuario ingrese su nombre de usuario y contraseña, todo esto a través de cuadros de texto y el campo de contraseña, como se observa en la Figura 50. El botón "Iniciar sesión" es la puerta de acceso a la ventana de Registro de medicamentos.

### Figura 50

*Diseño de la ventana inicio de sesión.*



INICIO DE SESIÓN



Usuario

Contraseña

**Iniciar sesión**

¿No tienes cuenta?

**Registrarse**

Recuerda no usar espacios en tu nombre de usuario

*Nota. Autoría*

Además del botón "Iniciar sesión", el dispensador también incluye un botón "Registrarse" para que los usuarios que no estén registrados puedan hacerlo. La ventana de registro, que se muestra en la Figura 51, solicita al usuario que proporcione su nombre de usuario y contraseña.



## Figura 51

### Programación de la ventana registro



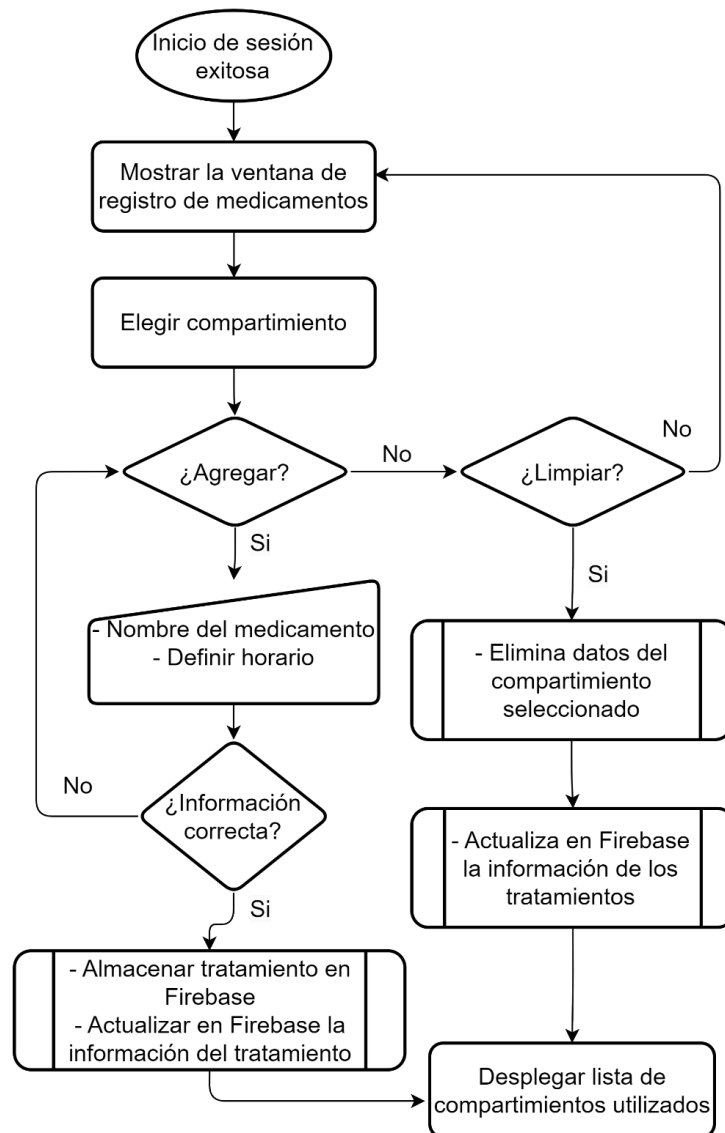
*Nota.* Autoría

### 3.5.7.2. Registro de Alarmas

Una de las características más importantes que debe tener el dispensador de pastillas, es la facilidad para la gestión del medicamento. Es por esta razón que se ha implementado una ventana que permita al usuario registrar y administrar sus pastillas. A través de esta ventana el usuario podrá registrar el nombre del medicamento, el horario en el que debe tomarlo y el compartimiento en el que se encuentra alojado. Esta información se guardará en la base de datos, para su sincronización con la Raspberry. En la Figura 52, se encuentra el diagrama de flujo del funcionamiento de esta ventana.

**Figura 52**

Diagrama de flujo del funcionamiento de la ventana registro de alarmas



*Nota.* Autoría

En cuanto al diseño, la pantalla incluye los campos que se ilustran en la Figura 53.

- **Compartimiento:** donde el usuario puede elegir el compartimiento donde se colocará la medicina.
- **Nombre de la medicina:** donde el usuario especificará el nombre de la medicina disponible en el compartimiento.

- **Hora de la toma:** campo disponible para que el usuario especifique el horario para que se libere la pastilla.
- **Cantidad de pastillas:** para que el usuario indique la cantidad de pastillas que se colocarán en el compartimiento seleccionado.
- **Lista:** en donde se muestran todos los compartimientos que han sido programados.
- **Botones agregar/ modificar y limpiar:** a través de estos elementos, el usuario podrá ejecutar las acciones de agregar el medicamento y limpiar los campos para el ingreso de nueva información.

**Figura 53**

*Diseño de la ventana registro de alarmas*

The image shows a mobile application interface for recording alarms. The title is "REGISTRO DE ALARMAS". It is divided into three main sections: "Medicamento", "Compartimiento", and "Horario".

- Medicamento:** Includes a pill icon and a text input field labeled "Nombre del medicamento".
- Compartimiento:** Includes a compartment icon, a label "Selecciona el compartimiento", and a dropdown menu labeled "Compartimiento". Below it is a label "Ingrese el número de pastillas" and another dropdown menu labeled "Cantidad".
- Horario:** Includes a clock icon and three dropdown menus for "Hora", "Min", and "Fran..". Below these is a label "Hora programada" followed by four dashes.

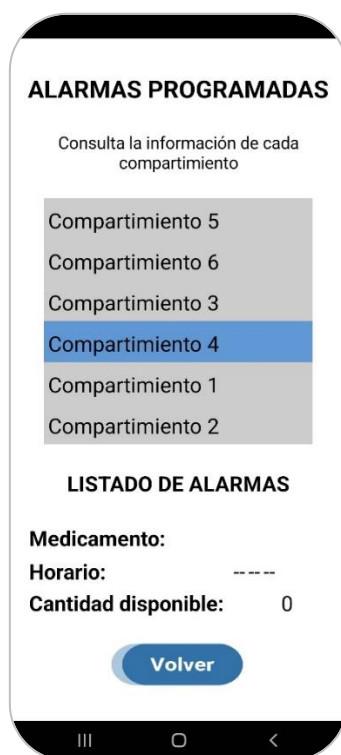
At the bottom, there are four buttons: "Registrar" (with a plus icon), "Limpiar" (with a trash icon), "Modo Viaje" (with a person icon), and "Alarmas Programadas" (with a list icon).

*Nota. Autoría*

La ventana "Alarmas programadas" muestra las alarmas que el usuario ha programado para cada compartimiento. Al seleccionar un compartimiento, se muestra la información almacenada para ese compartimiento. En la Figura 54 se ilustra el diseño de esta pantalla.

#### Figura 54

*Diseño de la ventana alarmas programadas*



*Nota. Autoría*

#### 3.5.7.3. Modo Viaje

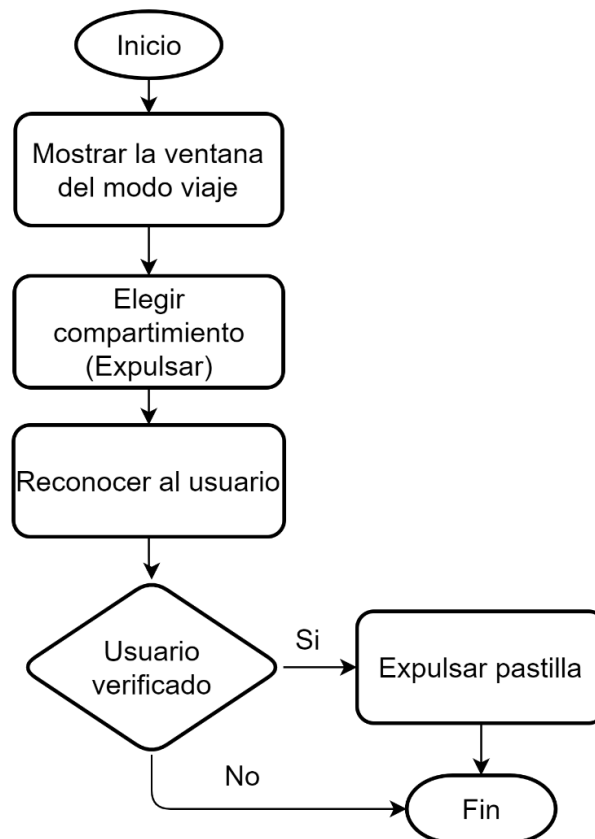
En ocasiones, el usuario tiene que ausentarse de casa. Sin embargo, al tratarse de un dispensador que tiene la necesidad de estar conectado ininterrumpidamente a la conexión eléctrica para funcionar, no es posible transportarlo de un lugar a otro. Para resolver este problema, se ha diseñado un modo de salida de casa que brinda al usuario la posibilidad de llevar sus medicamentos sin necesidad de pasar por el procedimiento de cumplimiento de la hora programada. Al presionar el botón de activación desde la aplicación móvil, el dispensador

recibe la señal, se verifica al usuario y el servo correspondientes se activa para girar y dispensar.

En la Figura 55 se ilustra el diagrama de flujo de esta ventana.

### Figura 55

*Diagrama de flujo del funcionamiento del modo viaje*



*Nota.* Autoría

Para la programación en App Inventor, se asignaron los bloques necesarios, como el botón de activación y selección del número de pastillas como se ilustra en la Figura 56. Para utilizar este modo, el usuario debe abrir la aplicación móvil, presionar el botón modo viaje y posteriormente el botón expulsar dependiendo del número de pastillas que desee retirar.

**Figura 56**

*Diseño de la ventana del modo viaje*



*Nota.* Elaborado por el autor

### **3.5.8. Diseño de Estructura**

Una de las mayores preocupaciones al diseñar los contenedores para pastillas es adaptarlos a la diversidad de formas y tamaño disponibles en el mercado. Por lo que se ha optado por construir contenedores tipo ruleta, estos ofrecen la flexibilidad necesaria para acomodar pastillas de distintas formas y tamaños.

La estructura física del dispensador de pastillas se diseñó para albergar todos los componentes, tanto electrónicos como mecánicos. Esto asegura la comodidad, la organización y la confiabilidad en la dispensación de medicamentos. La caja del dispensador tiene forma cúbica, como se observa en la Figura 57 y tiene las siguientes medidas: 30 cm de largo, 22,5

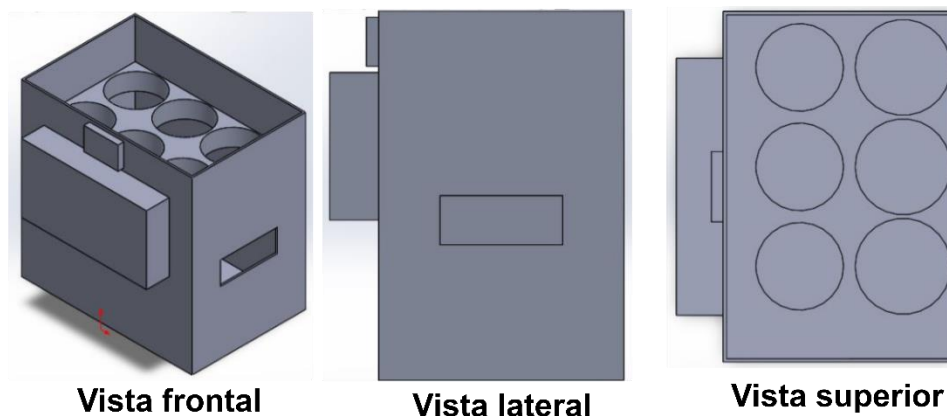
cm de alto y 20 cm de profundidad. Este tamaño es adecuado para su ubicación en diversos entornos.

- En la parte superior de la caja se encuentran distribuidos los contenedores de pastillas, que están dispuestos en una rueda que gira para dejar caer la pastilla.
- En la parte lateral izquierda se encuentra la bandeja en la cual caerán las pastillas que hayan sido dispensadas.
- En la parte frontal se encuentra distribuidas la cámara y la pantalla. La cámara se utiliza para reconocer el rostro del usuario, lo que ayuda a garantizar que solo el usuario autorizado pueda acceder al dispensador. La pantalla muestra información sobre el estado del dispensador, como la cantidad de medicamentos disponibles y el horario de las dosis.

Los componentes electrónicos se encuentran ubicados en la base del dispositivo para evitar el contacto entre ellos y los medicamentos

### **Figura 57**

#### *Estructura del dispensador de pastillas*



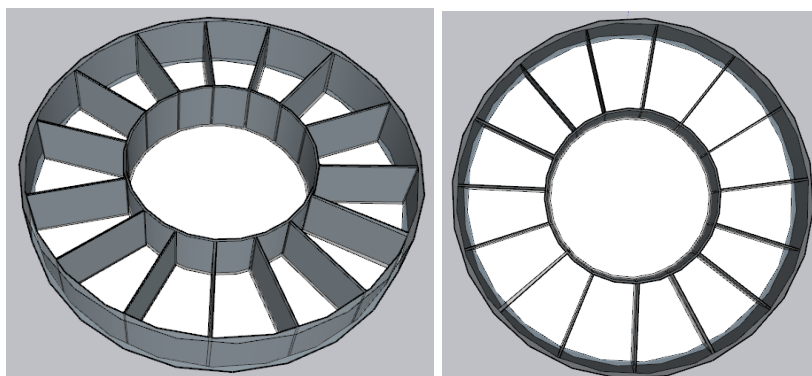
*Nota.* Autoría

Se diseñaron dos tipos de contenedores, cada uno compuesto por 15 secciones diseñadas para albergar una pastilla. Los contenedores de mayor diámetro, con 8 cm, están pensados para albergar píldoras, tabletas y comprimidos con formas circulares, mientras que los contenedores

de menor diámetro, con 7,5 cm están pensados para albergar formas farmacéuticas sólidas tipo cápsulas, píldoras, tabletas y comprimidos con formas rectangulares u ovaladas. El diseño de estos contenedores se ilustra en la Figura 58.

### **Figura 58**

#### *Estructura del contenedor de tabletas*



*Nota.* Autoría

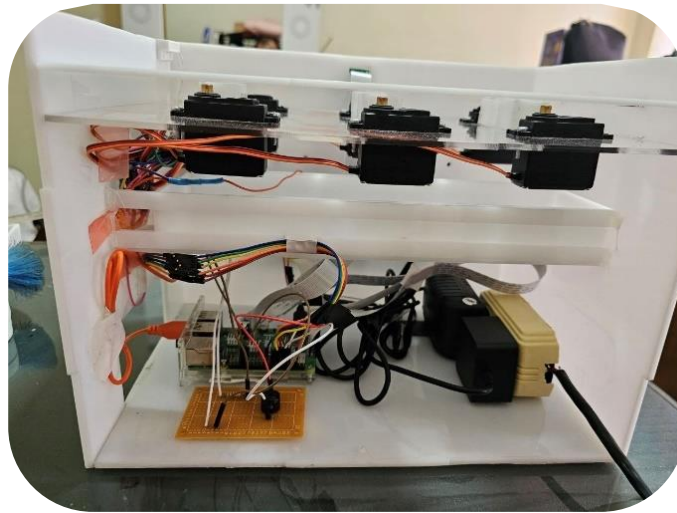
En cuanto al material seleccionado para la elaboración de la carcasa, se ha elegido el acrílico por sus cualidades y ventajas significativas. El acrílico es un polímero termoplástico altamente resistente que ofrece una serie de beneficios para el diseño del pastillero.

El acrílico es conocido por su durabilidad y rigidez, lo que garantiza una estructura sólida y confiable para proteger adecuadamente los medicamentos de la humedad y el polvo. En la Figura 59, se puede apreciar la estructura final del pastillero, en la que se muestra cómo el acrílico ha sido hábilmente moldeado y ensamblado para brindar un diseño funcional.



**Figura 59**

*Estructura en acrílico*



*Nota. Autoría*

Finalmente, se unieron cada uno de los subsistemas para obtener un prototipo funcional del dispensador de pastillas. En la Figura 60, se puede apreciar la vista general del prototipo completo, resultado de la integración de los diferentes componentes.

**Figura 60**

*Prototipo funcional*



*Nota. Autoría*

## Capítulo IV: Pruebas de Funcionamiento

Al concluir las fases de análisis de requerimientos y diseño del sistema, se da paso a la fase de pruebas en la cual se evalúa el funcionamiento general del prototipo dispensador de pastillas. En esta fase se evaluará si las acciones realizadas desde la aplicación son reflejadas de manera precisa en el dispensador, tanto para el registro de usuarios, medicamentos y horarios. También se pondrá a prueba aspectos como la fiabilidad de la liberación de pastillas y la generación de alertas. El objetivo es involucrar a usuarios reales, especialmente a adultos mayores, para probar el sistema en condiciones reales. Estas pruebas constituyen el último componente de la metodología de Diseño Centrado en el Usuario (DCU), marcando así el cierre de la integración de cada fase del proceso metodológico.

### 4.1. Cronograma de Pruebas de Funcionamiento del Sistema

El cronograma de pruebas ayuda a verificar cada funcionalidad del dispositivo, a su vez tomar el tiempo de respuesta de cada una de estas funcionalidades para comprobar que el dispositivo funciona de una manera óptima y eficiente, todo esto se puede visualizar en la Tabla 21, donde se distribuye cada una de estas actividades.

**Tabla 21**

*Cronograma de Pruebas*

<b>Cronograma de Pruebas</b>			
<b>Tipo de Prueba</b>	<b>Ubicación de la Prueba</b>	<b>Resultado Esperado</b>	<b>Duración</b>
<b>Prueba de registro y sincronización de la app móvil</b>	Entorno controlado	Obtener la creación de un perfil de usuario donde se puedan almacenar en la nube y sincronizarse con el dispensador.	1 día 14/07/2023
<b>Prueba de gestión de medicamentos desde la app móvil</b>	Entorno controlado	Verificar las funcionalidades agregar, modificar y eliminar medicamentos y horarios de toma de la medicación.	1 día 17/07/2023

<b>Prueba de tiempo de respuesta del dispensador</b>	Entorno controlado	Obtener resultados óptimos de tiempos de respuesta de cada proceso que se realiza en dispensar el medicamento en el prototipo.	3 días 18/07/2023 - 20/07/2023
<b>Prueba de usabilidad y experticia (Usuario 1)</b>	Vivienda del usuario 1	Obtener información de la adaptabilidad, facilidad y maniobrabilidad del dispositivo.	7 días 20/07/2023 - 26/07/2023
<b>Prueba de usabilidad y experticia (Usuario 2)</b>	Vivienda del usuario 2	Obtener información de la adaptabilidad, facilidad y maniobrabilidad del dispositivo.	7 días 27/07/2023 - 02/08/2023
<b>Prueba de usabilidad y experticia (Usuario 3)</b>	Vivienda del usuario 3	Obtener información de la adaptabilidad, facilidad y maniobrabilidad del dispositivo.	7 días 03/08/2023 - 09/08/2023

---

*Nota.* Autoría

Los tiempos establecidos para cada una de las pruebas fueron pensados para garantizar que se cumplan todos los objetivos planteados, desde un punto de vista crítico como técnico. Se asignó un día de prueba para verificar la creación usuarios, enviar esta información a la nube y sincronizarla con el dispositivo. Un día adicional se utilizará para verificar el uso de la aplicación para manipular los horarios de distribución de pastillas. Una vez que se completen estos procesos, se comenzará con las pruebas de tiempo de respuesta el cual tendrá una duración de 3 días. Estas pruebas deben tener en cuenta cada uno de los procesos para la dispensación del medicamento y los tiempos de respuesta de cada uno de ellos. Por último, las pruebas de usabilidad y experiencia del usuario que tendrán una duración de una semana para uno de los usuarios y se utilizarán para evaluar la maniobrabilidad básica de la tecnología y la conectividad a Internet para la manipulación del dispensador.

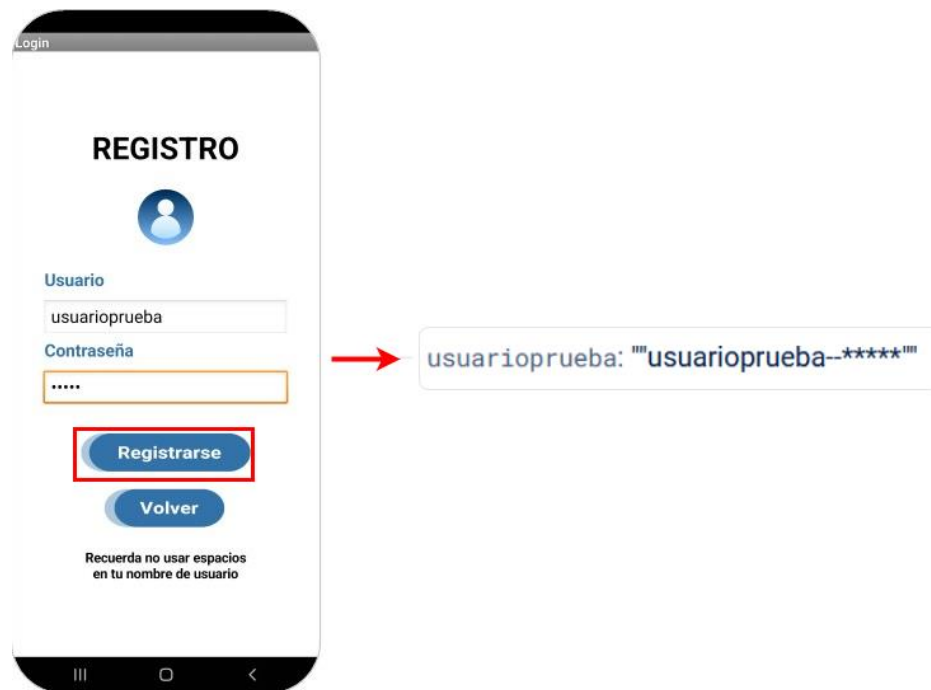
A continuación, se describen con más detalle cada una de las pruebas:

- **Creación de usuario:** Esta prueba se utilizará para verificar que el proceso de creación de un usuario sea sencillo y rápido. El usuario debe poder crear una cuenta, ingresar sus datos personales y sincronizar su cuenta con la nube.

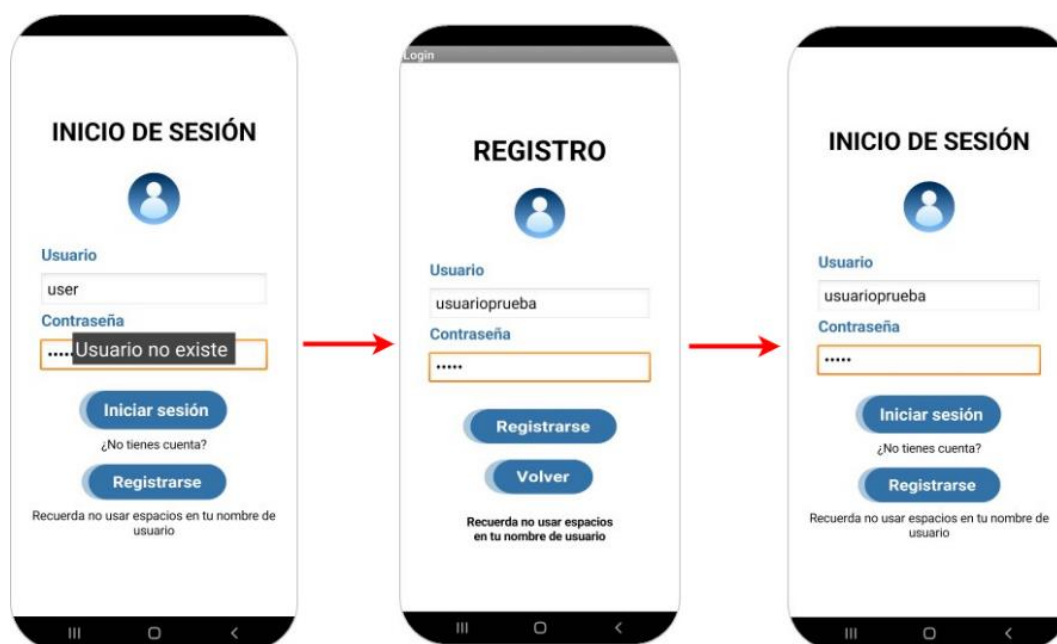
- **Verificación del uso de la aplicación:** Esta prueba se utilizará para verificar que el usuario pueda usar la aplicación para manipular los horarios de distribución de pastillas. El usuario debe poder agregar nuevos medicamentos, cambiar los horarios de distribución y eliminar medicamentos.
- **Pruebas de tiempo de respuesta:** Estas pruebas se utilizarán para verificar que los tiempos de respuesta para la dispensación de medicamentos sean aceptables. El usuario debe poder recibir sus medicamentos en un tiempo razonable, independientemente de la cantidad de medicamentos que esté dispensando.
- **Pruebas de usabilidad y experiencia del usuario:** Estas pruebas se utilizarán para evaluar la maniobrabilidad básica de la tecnología y la conectividad a Internet para la manipulación del dispensador. El usuario debe poder usar la tecnología de manera fácil y rápida, incluso si no tiene experiencia previa con ella.

#### **4.2. Prueba de Registro y Sincronización de la Aplicación Móvil**

Con esta prueba se pretende verificar si es posible crear cuentas en la aplicación móvil de manera exitosa, así como también que los datos ingresados se almacenen de manera adecuada en la base de datos de Firebase y que a su vez los cambios realizados en la aplicación se sincronicen con el dispensador. El primer paso será tener la aplicación móvil instalada en el teléfono. El siguiente paso será crear registrar un nuevo usuario “usuarioprueba”, posteriormente se verifica que no se pueda volver a ingresar ese usuario ya registrado. Inmediatamente se revisa el estado de la base de datos, para corroborar que se haya registrado con éxito al nuevo usuario “usuarioprueba”, en la Figura 61, se puede visualizar al nuevo usuario registrado.

**Figura 61***Registro de usuario**Nota. Autoría*

Posteriormente se intentará iniciar sesión con este usuario, pero antes se probará el ingreso con un usuario no registrado. En la Figura 62, se verifica este proceso. Al ingresar un usuario no registrado se despliega un mensaje de “Usuario no existe” y finalmente cuando se prueba con las credenciales correctas, se despliega la siguiente ventana de registro de medicamentos.

**Figura 62***Inicio de sesión del usuarioprueba**Nota. Autoría*

### 4.3. Prueba de Gestión de Medicamentos Desde la Aplicación Móvil

Esta prueba se enfoca en evaluar la funcionalidad y eficiencia de la aplicación móvil para permitirle al usuario agregar, modificar y eliminar medicamentos y horarios de toma de la medicación. El paso previo para realizar esta prueba es verificar que el dispensador esté correctamente cargado con las pastillas y que la aplicación esté instalada y configurada en el dispositivo móvil.

#### 4.3.1. Agregar Nuevo Medicamento

Para comprobar esta funcionalidad, es necesario desplazarse hasta la ventana de registro de medicamentos y prestar atención al botón agregar/modificar desplegado en la interfaz. Se utiliza los campos disponibles y se ingresa información como el compartimiento a utilizar, nombre del medicamento y el horario de administración. Posteriormente, se verifica que el medicamento se agregue correctamente y se muestre en la lista de medicamentos. Además, se comprueba que esta información se haya guardado en la base de datos. En la Figura 63, puede

observar la información guardada tanto del compartimiento, el nombre del medicamento y el horario programado.

**Figura 63**

*Agregar nuevo medicamento*



*Nota. Autoría*

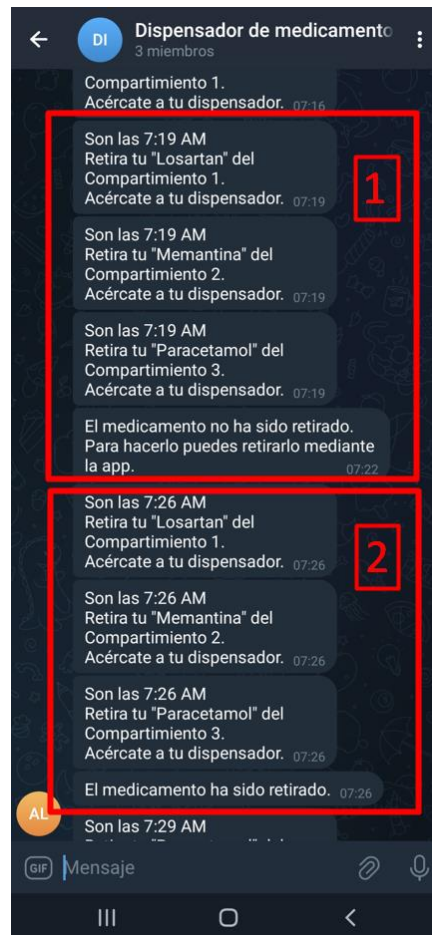
#### 4.3.3. Verificar Notificaciones Generadas

Para realizar esta prueba, se configura una alarma desde la aplicación, cuando se cumpla el horario establecido, se mostrará una notificación en la pantalla del dispositivo móvil del usuario.

En el escenario 1, el usuario no se acerca al dispensador y no verifica su identidad. Después de transcurridos 3 minutos, se enviará una notificación para indicar que no se ha retirado el medicamento y que se lo puede hacer a través de la aplicación utilizando el modo viaje. En el escenario 2, el usuario se acerca al dispensador y verifica su identidad. Una vez que el usuario se ha identificado, el dispensador activa el servomotor que abre el compartimiento correspondiente. El medicamento se dispensa y se envía una notificación al dispositivo móvil del usuario para indicar que el medicamento ha sido retirado. Estas notificaciones se muestran en la Figura 64.

**Figura 64**

*Alertas generadas a través del bot de Telegram*



*Nota. Autoría*

#### **4.4. Tiempo de Respuesta del Dispensador de Pastillas**

Estas pruebas se utilizarán para verificar que los tiempos de respuesta para la dispensación de medicamentos sean aceptables. El usuario debe poder recibir sus medicamentos en un tiempo razonable, independientemente de la cantidad de medicamentos que esté dispensando.

##### **4.4.1. Tiempo de Entrega de Medicamento**

Desde la aplicación se configuraron diferentes compartimientos para que se entreguen pastillas en intervalos aleatorios de tiempo, con la intención de medir el tiempo empleado desde que el usuario es reconocido por la cámara hasta que el medicamento esté listo para ser



recogido. La precisión de esta prueba depende de factores como la precisión del reconocimiento y velocidad de respuesta de los servos. En la Tabla 22 se detallan los resultados obtenidos.

**Tabla 22**

*Tiempo de dispensación registrado*

Nro.	Compartimiento	Tiempo (s)
1	1	8,68
2	2	8,26
3	3	8,17
4	4	7,95
5	5	8,35
6	6	7,96
7	1	8,81
8	2	9,14
9	3	8,46
10	4	8,51
11	5	8,89
12	6	8,03

*Nota.* Autoría

Se tomaron 12 muestras para un obtener un conjunto de datos suficiente para realizar el análisis. Partiendo de estos valores se obtiene un tiempo máximo de 9 segundos y un tiempo mínimo de 7 segundos para la entrega del medicamento. El cálculo del tiempo promedio de entrega del medicamento se calcula utilizando la Ecuación X, donde t es el tiempo de cada una de las pruebas y n el número de muestras tomadas.

$$\text{Tiempo promedio} = \bar{t} = \frac{\sum_{i=1}^n t_i}{n}$$

$$\bar{t} = \frac{101,41}{12}$$

$$\bar{t} = 8,45 \text{ s}$$

Se obtuvo un tiempo promedio de 8,43 segundos que el dispensador requiere para dispensar una pastilla. Como una forma de obtener una métrica más completa del rendimiento de las pruebas de tiempo de entrega del medicamento, se calcula la desviación estándar con la Ecuación X, donde  $t_i$  es el tiempo de cada muestra. A través de este valor se puede conocer la variabilidad de los resultados de prueba y a su vez la fiabilidad del dispensador.

$$\text{Desviación estándar} = \sigma = \sqrt{\frac{\sum_1^n (t_i - \bar{t})^2}{n}}$$

$$\sigma = \sqrt{\frac{1,6169}{12}}$$

$$\sigma = 0,3648$$

Se considera que no hay un valor óptimo específico para la desviación estándar que se pueda aplicar a todos los procesos, pues este valor depende del contexto en el que esté el producto. Sin embargo, hay pautas generales que pueden ser tomadas en consideración para evaluar un determinado proceso, estas pautas se muestran en la Tabla 23.

### Tabla 23

*Pautas para el establecimiento de un valor óptimo de desviación estándar*

Requerimientos del proceso	% de desviación estándar
Alta precisión – Baja variabilidad	5% al 10%
Precisión moderada	10% al 20%
Flexibilidad en la precisión	Hasta el 30%

Necesidad de optimización

Mayor al 30%

*Nota. Autoría*

El porcentaje de desviación estándar se calcula dividiendo la desviación estándar entre la media y multiplicando el resultado por 100 para obtener el valor en porcentaje.

$$\% \sigma = \frac{\sigma}{\bar{t}}$$

$$\% \sigma = \frac{0,3648}{8,45} \times 100$$

$$\% \sigma = 4,31\%$$

La desviación estándar para el tiempo de entrega del medicamento fue de 4,31%, al comparar este valor con los de la pauta general, resulta en que este proceso del sistema es óptimo.

#### **4.4.2. Tiempo de Entrega de Notificaciones**

Con esta prueba se pretende verificar que las notificaciones y recordatorios se envíen correctamente a la aplicación móvil para recordar al usuario sobre las tomas de medicamentos. Para verificar y evaluar el tiempo que transcurre entre el horario programado para la toma del medicamento y la alerta entregada por el bot de Telegram en el teléfono celular, se programó una serie de horarios aleatorios de tiempo para la toma del medicamento. Para esta prueba es necesario prestar atención en la hora definida para la entrega del medicamento y registrar la hora exacta en la que se entrega la alerta. Con la diferencia entre estos dos parámetros se puede analizar el tiempo promedio de entrega de alertas.

#### **Tabla 24**

*Tiempo de entrega de notificaciones a través del bot de Telegram*

<b>Nro.</b>	<b>Compartimiento</b>	<b>Hora programada</b>	<b>Hora entrega alerta</b>
-------------	-----------------------	------------------------	----------------------------

1	1	8:00 am	8:00 am
2	2	10:00 am	10:00 am
3	3	10:30 am	10:30 am
4	4	11:00 am	11:00 am
5	5	11:15 am	11:15 am
6	6	12:00 pm	12:00 pm
7	1	12:45 pm	12:45 pm
8	2	1:00 pm	1:00 pm
9	3	2:00 pm	2:00 pm
10	4	2:30 pm	2:30 pm
11	5	3:00 pm	3:00 pm
12	6	5:00 pm	5:00 pm

*Nota. Autoría*

El tiempo registrado en la Tabla 24 se basa en el tiempo registrado por el bot de Telegram. Cuando la notificación llega al teléfono celular, el bot registra la hora en la que llegó la notificación. Este tiempo se compara con el tiempo registrado en el dispensador para determinar si hay un retraso. Los resultados de la comparación muestran que no hay retrasos que excedan el minuto, puesto que en el momento en el que el programa detecta una coincidencia de horario, se envía la notificación.

#### ***4.4.3. Precisión en la Dispensación***

Para esta prueba se han definido horarios de prueba específicos, para simular diferentes momentos de medicación. Los parámetros de precisión de la alerta y precisión en la dispensación, es decir, que el medicamento que se dispensa corresponde al medicamento programado serán registrados en la Tabla 25.

**Tabla 25***Resultados prueba de precisión*

<b>Horario programado</b>	<b>Medicamentos programados</b>	<b>Hora de alerta</b>	<b>Medicamentos entregados</b>
7:30	1,2,4	7:30	1,2,4
8:45	2,6,5	8:45	2,6,5
9:30	1,2,3,4	9:30	1,2,3,4
12:00	1	12:00	1
12:45	4,3,2,1	12:45	4,3,2,1
13:15	2	13:15	2
15:00	4,5,2	15:00	4,5,2
15:30	5	15:30	5
16:15	5,1	16:15	5,1
17:00	6,3,4	17:00	6,3,4

*Nota.* Autoría

En la Tabla 25, se evidencia que se ha logrado un cumplimiento del 100% en los horarios y una precisión total en la dispensación de medicamentos.

#### **4.5. Prueba de Usabilidad y Experiencia de Usuario**

Con esta prueba se pretende evaluar cómo los usuarios interactúan con el dispensador de pastillas y la aplicación de gestión. Para esto se ha seleccionado y solicitado la colaboración de tres adultos mayores, con los que se realizará un seguimiento durante un periodo de tres semanas, distribuidas en intervalos de tiempo de una semana para cada uno. Todos estos usuarios toman medicamentos diariamente para tratar sus enfermedades y cada uno tiene diferentes niveles de experiencia con el uso de dispositivos electrónicos, desde aquellos que

son muy hábiles con los dispositivos digitales, hasta aquellos que tienen poca experiencia. Esta diversidad brinda la posibilidad de tener una visión más completa de cómo diferentes adultos mayores pueden interactuar con el dispensador y la aplicación móvil.

Durante el periodo de prueba, los usuarios serán guiados en el uso del dispensador de medicamentos y la aplicación móvil como parte de su rutina diaria de toma de medicamentos. Se les solicitará que programen las alarmas para la toma de sus medicamentos y a la par que brinden una retroalimentación sobre su experiencia en general, incluyendo cualquier dificultad que hayan encontrado durante su uso.

Los datos recolectados durante la prueba incluyen la frecuencia con la que los usuarios utilizan el dispensador de pastillas y la aplicación móvil. El tiempo que tardan los usuarios en completar las tareas relacionadas con el dispensador de pastillas y los comentarios de los usuarios sobre su experiencia.

#### **4.5.1. Prueba Usabilidad: Caso 1**

Este usuario ha sido denominado “Usuario N01”, se trata de un adulto mayor de 75 años que consume diariamente cinco medicamentos diferentes. De acuerdo con la información registrada en la Tabla 26, la mayor parte de sus medicamentos deben ser tomados por las mañanas. Este usuario en particular opta por tomar sus pastillas en intervalos de una hora.

**Tabla 26**

*Perfil del usuario 1 – Caso 1*

<b>Nombre</b>	<b>Edad</b>	<b>Discapacidad</b>	<b>Medicamento</b>	<b>Dosis</b>	<b>Hora</b>
Usuario N01	73	No	Esomeprazol	1 diaria	6 am
			Prednisona	1 diaria	8 am
			Memantina	1 diaria	8 am
			Complejo B	1 diaria	9 am
			Alopurinol	1 diaria	10 am

*Nota. Autoría*

En la Figura 65 se puede observar la interacción del usuario 1 con el dispensador de medicamentos.

### **Figura 65**

*Interacción del usuario 1 con el dispensador*



*Nota. Autoría*

La Tabla 27 se utiliza para registrar el desempeño del dispensador durante las pruebas realizadas con el usuario 1. La tabla incluye cuatro columnas que recolectarán los datos de los siguientes parámetros:

- Reconocimiento del usuario (1/0): indica si el dispensador reconoció correctamente la identidad del usuario.
- Generación de alerta (1/0): indica si el dispensador generó una alerta para recordarle al usuario que se cumplió la hora para la toma del medicamento.

- Entrega de medicamento (1/0): para indicar si el dispensador entregó el medicamento.
- Medicamento correcto (1/0): para indicar que si el dispensador entregó la dosis correcta de medicamento.

Con el fin de realizar una evaluación objetiva del funcionamiento del dispensador, por lo que la valoración se ha realizado en términos de 1 y 0, donde el 1 indica que se cumplió el parámetro evaluado, por otro lado, el 0 indica que el dispensador no cumplió con el parámetro. Esto se realiza en base a cada una de las alarmas diarias configuradas, que para este caso son 6. Al finalizar se realiza una sumatoria de todos los valores obtenidos en cada parámetro, la sumatoria brindará una perspectiva general de cómo está funcionando el dispensador.

**Tabla 27**

*Resultados de la interacción del usuario 1 – caso 1*

<b>Día</b>	<b>Reconocimiento del usuario /6</b>	<b>Generación de alertas /6</b>	<b>Entrega de medicamento /6</b>	<b>Medicamento Correcto /6</b>
1	6	6	6	6
2	6	6	6	6
3	6	6	6	6
4	5	6	5	5
5	5	6	5	5
6	6	6	5	5
7	5	6	5	5
Valoración 42 puntos	39	42	39	39

*Nota. Autoría*



#### 4.5.2. Prueba Usabilidad: Caso 2

El usuario del caso 2 ha sido denominado “Usuario N02”, se trata de un adulto mayor de 71 años que consume diariamente seis medicamentos diferentes. De acuerdo con la información registrada en la Tabla 28, la mayor parte de sus medicamentos deben ser tomados por las mañanas. Este usuario en particular toma su medicación de la mañana, después del desayuno, lo hace paulatinamente en el horario de 10 am.

**Tabla 28**

*Perfil del usuario 2 – Caso 2*

Nombre	Edad	Discapacidad	Medicamento	Dosis	Hora
Usuario N02	70	No	Losartan	1 diaria	10 am
			Clopidogrel	1 diaria	10 am
			Memantina	1 diaria	10 am
			Sertralina	1 diaria	2 pm
			Alopurinol	1 diaria	10 am
			Complejo B	1 diaria	10 am

*Nota.* Autoría

En la Figura 66 se puede observar la interacción del usuario 2 con el dispensador de medicamentos.

**Figura 66**

*Interacción del usuario 2 con el dispensador*



En la Tabla 29 se recopilan los datos referentes a los aciertos o fallos generados por las diferentes funcionalidades del dispensador.

**Tabla 29**

*Resultados de la interacción del usuario 2 – caso 2*

<b>Día</b>	<b>Reconocimiento del usuario /6</b>	<b>Generación de alertas /6</b>	<b>Entrega de medicamento /6</b>	<b>Medicamento Correcto /6</b>
1	6	6	6	6
2	6	6	6	6
3	5	6	5	5
4	5	6	5	5
5	6	6	6	6
6	6	6	6	6
7	6	6	6	6
Valoración 42 puntos	40	42	40	40

*Nota. Autoría*

#### **4.5.3. Prueba Usabilidad: Caso 3**

El usuario del caso 3 ha sido denominado “Usuario N03”, se trata de un adulto mayor de 70 años que consume diariamente seis medicamentos diferentes. De acuerdo con la información registrada en la Tabla 30, sus medicamentos deben ser tomados en la mañana, en la tarde y mayormente en la noche.

**Tabla 30**

*Perfil del usuario 3 – Caso 3*

<b>Nombre</b>	<b>Edad</b>	<b>Discapacidad</b>	<b>Medicamento</b>	<b>Dosis</b>	<b>Hora</b>
Usuario N03	S	No	Irbesartán	1 diaria	9 am
			Levotiroxina	1 diaria	7 am
			Memantina	1 diaria	8 am
			Sertralina	1 diaria	2 pm
			Trayenta	1 diaria	8 pm
			Atorvastatina	1 diaria	8 pm

*Nota. Autoría*

En la Figura 67 se puede observar la interacción del usuario 3 con el dispensador de medicamentos.

**Figura 67***Interacción de usuario 3 con el dispensador*

*Nota.* Autoría

En la Tabla 31 se recopilan los datos referentes a los aciertos o fallos generados por las diferentes funcionalidades del dispensador.

**Tabla 31***Resultados de la interacción del usuario 3 – caso 3*

<b>Día</b>	<b>Reconocimiento del usuario /6</b>	<b>Generación de alertas /6</b>	<b>Entrega de medicamento /6</b>	<b>Medicamento Correcto /6</b>
1	6	6	6	6
2	5	6	5	5
3	6	6	6	6
4	6	6	6	6
5	6	6	6	6
6	6	6	6	6
7	6	6	6	6

Valoración 42 puntos	41	42	41	41
-------------------------	----	----	----	----

---

*Nota.* Autoría

## 4.6. Resultados

### 4.6.1. Resultados Cuantitativos

Los datos numéricos y las mediciones precisas son necesarios para evaluar de forma objetiva el rendimiento del dispensador y la aplicación móvil. La Tabla 32 muestra un resumen detallado del desempeño del dispensador en cuanto al cumplimiento de horarios, la efectividad del reconocimiento facial y la entrega de notificaciones al usuario. Este análisis incluye tanto los aciertos como los fallos, lo que proporciona una visión completa de la eficacia del sistema en estas áreas.

**Tabla 32**

*Rendimiento general de las funciones del dispensador*

Usuario	Reconocimiento del usuario	Generación de alertas	Entrega de medicamento	Medicamento Correcto
<b>Usuario 1</b>	39	42	39	39
<b>Usuario 2</b>	40	42	40	40
<b>Usuario 3</b>	41	42	41	41
<b>Valoración 126 puntos</b>	120	126	120	120

---

- Precisión en la Administración de Medicamentos.** La capacidad del dispensador para mantenerse fiel a los horarios programados es esencial para asegurar que los usuarios reciban su medicación en el momento adecuado. En este sentido, el dispositivo demostró una precisión del 95,24 % en la distribución correcta de pastillas según los horarios programados a través de la aplicación móvil. Este nivel de precisión es un indicador de que el dispensador es confiable para la administración de medicamentos.

- **Eficacia del Reconocimiento Facial.** El sistema logró identificar a los usuarios autorizados con una precisión del 100 % en condiciones de iluminación óptimas. Sin embargo, en condiciones de poca luz, la precisión del sistema se redujo a un 95,24 %. Esta disminución en la precisión sugiere que el sistema debe colocarse en un área bien iluminada para mejorar su rendimiento.
- **Entrega de Notificaciones.** El sistema logró entregar las notificaciones a través del bot de Telegram al usuario con una precisión del 100%. Los recordatorios puntuales y confiables para tomar la medicación de acuerdo con los horarios programados son posibles gracias a esta alta precisión.

#### **4.6.2. Resultados Cualitativos**

- **Experiencia con la Aplicación Móvil.** La interacción con la aplicación móvil fue desafiante para uno de los tres usuarios de prueba. Dos de los usuarios pudieron interactuar con la aplicación y la describieron como “fácil de usar”. Sin embargo, el tercer usuario tuvo complicaciones debido a su falta de experiencia en el uso de aplicaciones móviles y problemas de visión. Estos resultados eran predecibles debido a que este grupo etario está poco familiarizado con el uso de la tecnología y en sí de las aplicaciones móviles. Sin embargo, la ventaja clave destacada por todos los usuarios fue la capacidad de programar la aplicación de forma remota, ya que, si encontraban dificultades con la aplicación, un familiar o cuidador podía ayudarlos desde cualquier ubicación con acceso a Internet.

## Conclusiones y Recomendaciones

### Conclusiones

La revisión del estado del arte de los dispensadores de medicamentos disponibles en el mercado reveló que, si bien ofrecen soluciones útiles, a menudo carecen de la capacidad de reconocer las necesidades individuales de los usuarios y adaptarse a los diferentes tamaños de medicamentos, una implicación especialmente relevante para las personas mayores, que suelen tomar una amplia gama de medicamentos con formas y tamaños variados.

La determinación de los requerimientos para el dispensador automático de pastillas utilizando una metodología de Diseño Centrada en el Usuario ha sido un paso fundamental en este proyecto. Al examinar de cerca las necesidades específicas de las personas mayores y considerar su comodidad y seguridad, se ha asegurado que el dispositivo sea funcional y adecuado para su uso diario.

El diseño del software, empleando un lenguaje de programación como Python ha permitido responder a las necesidades y problemas planteados en el control del sistema y el interfaz de usuario. Su simplicidad y el potencial que presenta lo convierten en un lenguaje amigable para el desarrollador, pero sin perder ni un ápice de competencias.

La fase de implementación ha sido fundamental en la creación de nuestro dispensador inteligente de pastillas. Se logró interconectar con éxito el sistema electrónico, mecánico y de reconocimiento con una aplicación móvil. Esto permitirá a los usuarios gestionar las acciones del dispensador de forma remota y recibir alertas importantes para el seguimiento de su medicación.

El prototipo de dispensador automatizado de pastillas ha demostrado el rendimiento esperado al superar con éxito una serie de pruebas. Estas pruebas abarcaron desde la entrega precisa de medicamentos en horarios programados hasta la efectividad del sistema de reconocimiento facial en diversas condiciones de iluminación. Los resultados de estas pruebas

confirman que el dispensador es un dispositivo de alta calidad y confiable que cumple con su propósito de proporcionar una gestión eficaz de la medicación para las personas de la tercera edad. Además, se ha demostrado que el diseño centrado en el usuario ha contribuido significativamente a su facilidad de uso y accesibilidad, lo que lo convierte en una herramienta valiosa para mejorar la calidad de vida de los usuarios.



## Recomendaciones

En futuras iteraciones, se recomienda adaptarle al dispensador un tipo de batería o banco de baterías que abastezca por un largo período de tiempo. Esto aumentaría significativamente la autonomía del dispensador, permitiéndole funcionar durante cortes de energía prolongados. De esta forma, se garantizaría la continuidad en la administración de medicamentos, incluso en situaciones de emergencia

La aplicación móvil, inicialmente implementada en el sistema operativo Android, puede expandirse a iOS para aumentar su accesibilidad en diversos smartphones, con dicha estrategia de desarrollo multiplataforma permitirá que un público más amplio, independientemente del sistema operativo de sus dispositivos móviles, pueda aprovechar las ventajas y la funcionalidad que la aplicación proporciona.

Se sugiere incorporar sensores adicionales al prototipo para obtener una visión más completa del estado de salud del usuario. Estos sensores podrían medir parámetros como la temperatura corporal, la frecuencia cardíaca o la presión arterial. La información recopilada de estos sensores podría ser de gran utilidad para los cuidadores y profesionales de la salud, ya que les proporcionaría una visión más detallada de la condición del usuario. Esto podría resultar en una atención médica más precisa y oportuna.

Se recomienda instalar el dispensador a una altura adecuada para las personas mayores, teniendo en cuenta sus limitaciones de movilidad. Además, es importante garantizar condiciones óptimas de iluminación en el lugar donde se coloque el dispensador. Una iluminación adecuada facilita la interacción con la pantalla táctil y el sistema de reconocimiento facial, lo que contribuye a una experiencia de uso segura y cómoda

## Bibliografía

- Achulli, J. (2022). *El desarrollo de las aplicaciones móviles*.
- Agrawal, A. K., & Singh, Y. N. (2015). Evaluation of face recognition methods in unconstrained environments. *Procedia Computer Science*, 48(C), 644–651.  
<https://doi.org/10.1016/j.procs.2015.04.147>
- Alvarez, I. (2020). *Arquitectura hardware de un sistema de reconocimiento utilizando histogramas de gradientes orientados y una red neuronal DEN*. Universidad Tecnológica de la Mixteca.
- Amazon. (n.d.). *¿Qué es un IDE?* <https://aws.amazon.com/es/what-is/ide/>
- Anwarul, S., & Dahiya, S. (2020). A Comprehensive Review on Face Recognition Methods and Factors Affecting Facial Recognition Accuracy. *Proceedings of ICRIC 2019: Recent Innovations in Computing*, 495–514.
- Asamblea Nacional del Ecuador. (2019). Ley Orgánica De Las Personas Adultas Mayores. *Suplemento Del Registro Oficial No. 484, 484*.
- Banco Mundial. (2022). *Población total*.  
<https://datos.bancomundial.org/indicador/SP.POP.TOTL>
- Barreto, J. (2019). *Prototipo de dispensador de medicamentos automáticos para personas de la tercera edad*. Universidad El Bosque.
- Basañez, D. (2018). *Sistema de Reconocimiento Facial*. Universidad del País Vasco.
- Bello, N., & Montoya, P. (2017). Adherencia al tratamiento farmacológico en adultos mayores diabéticos tipo 2 y sus factores asociados. *Gerokomos*, 28(2), 73–77.
- Cárdenas, I., Librada, M., & Mesa, C. (2019). *Las generaciones digitales y las aplicaciones móviles como refuerzo educativo*. 2, 27–28. <http://remca.umet.edu.ec/>
- Chandi, L. (2017). *Procesos de Desarrollo de Software en Aplicaciones Móviles*.

- Cheney, D., & Hall, A. (2022, October 24). *Revisión del dispensador de píldoras Hero*.  
<https://www.forbes.com/health/healthy-aging/hero-pill-dispenser-review/>
- Clark, A. (2023, January 27). *Revisión del dispensador de medicamentos Pria*.  
<https://www.theseniorlist.com/medication/dispensers/pria/>
- Dilla, T., Valladares, A., Lizán, L., & Sacristán, J. A. (2009). Adherencia y persistencia terapéutica: causas, consecuencias y estrategias de mejora. *Atención Primaria, 41*(6).  
<https://doi.org/10.1016/j.aprim.2008.09.031>
- Domínguez, S. (2017). *Reconocimiento facial mediante el Análisis de Componentes Principales (PCA)*. Universidad de Sevilla.
- Figuroa, C., Otzen, T., Figuroa, V., Sanhueza, P., & Candía, J. (2020). Alfabetización en salud y adherencia farmacológica en adultos mayores con enfermedades crónicas, de la ciudad de Temuco. *Revista Médica de Chile, 148*(5).
- Filié Haddad, M., Satie Takamiya, A., Martins da Silva, E. M., & Barros Barbosa, D. (2009). Farmacología en la tercera edad: medicamentos de uso continuo y peligros de la interacción medicamentosa. *Gerokomos, 20*(1). <https://doi.org/10.4321/s1134-928x2009000100004>
- García, C., Mondragón, J., & Rojas, E. (2022). *Prototipo de dispensador de comprimidos y cápsulas medicinales configurable y administrable desde una aplicación móvil*. Instituto Politécnico Nacional.
- Garreta, M., & Mor, E. (2020). *Diseño centrado en el usuario* (pp. 39–43).  
[http://cv.uoc.edu/annotation/988e6d1ead39c2f9c461c07b3ce0e9e7/754931/PID\\_00275324/PID\\_00275324.html](http://cv.uoc.edu/annotation/988e6d1ead39c2f9c461c07b3ce0e9e7/754931/PID_00275324/PID_00275324.html)
- Gironés, J. (2019). *El gran libro de Android* (Alpha Editorial, Ed.).

- Gray, S. L., Mahoney, J. E., & Blough, D. K. (2001). Medication adherence in elderly patients receiving home health services following hospital discharge. *Annals of Pharmacotherapy*, 35(5). <https://doi.org/10.1345/aph.10295>
- Hawkins, M. (2018). *Guía simple para el encabezado GPIO de Raspberry Pi*. <https://www.raspberrypi-spy.co.uk/2012/06/simple-guide-to-the-rpi-gpio-header-and-pins/>
- Hernández, I., Sarmiento, N., Gonzales, I., Galarza, S., De la Bastida, A., Terán, S., & Terán, E. (2018). Adherencia al tratamiento en los pacientes de consulta externa de los centros de salud de Quito. *Metro Ciencia*, 7–11.
- Ibrahim, D. (2021). ARM Cortex microcontroller development boards. In *Arm-Based Microcontroller Multitasking Projects* (pp. 33–45). Elsevier. <https://doi.org/10.1016/B978-0-12-821227-1.00003-7>
- Intel. (2021, October 28). *Diferentes protocolos de Wi-Fi y velocidades de datos*. <https://www.intel.la/content/www/xl/es/support/articles/000005725/wireless/legacy-intel-wireless-products.html>
- Intel. (2023). *Referencia de chipset y sistema en un chip (SoC) para Intel® NUC*. <https://www.intel.com/content/www/us/en/support/articles/000056236/intel-nuc.html>
- ISO/IEC/IEEE. (2018). *Norma internacional ISO/IEC/IEEE - Ingeniería de sistemas y software - Procesos del ciclo de vida - Ingeniería de requisitos*.
- Jones, M. (2016). *MG995 Servoactuador de alta velocidad*. <https://www.alldatasheet.es/datasheet-pdf/pdf/1132435/ETC2/MG995.html>
- Kortli, Y., Jridi, M., Al Falou, A., & Atri, M. (2020). Face recognition systems: A survey. *Sensors*, 20(2). <https://doi.org/10.3390/s20020342>

- López-Pérez, B., Fernández-Pinto, I., & Márquez-González, M. (2017). Educación emocional en adultos y personas mayores. *Electronic Journal of Research in Education Psychology*, 6(15). <https://doi.org/10.25115/ejrep.v6i15.1286>
- MecatrónicaLATAM. (2021). *Servomotor*.  
<https://www.mecatronicalatam.com/es/tutoriales/motor/motores-electricos/motor-de-corriente-continua/servomotor/>
- MedMinder. (2023). *Dispensador Automático de Pastillas*.  
<https://www.medminder.com/automatic-pill-dispenser/>
- Mena, M. (2023, March 30). *El mapa mundial de Android e iOS*. Industria y Consumo Mundial de Smartphones. <https://es.statista.com/grafico/29620/sistema-operativo-movil-con-la-mayor-cuota-de-mercado-por-pais/>
- Moscoso, X., & Villacres, F. (2020). *Diseño e implementación de un prototipo de maquina dispensadora de medicamentos en forma de pastilla para personas de la tercera edad que padecen enfermedades no transmisibles*.
- Najafi Khanbebin, S., & Mehrdad, V. (2021). Enfoque de mejora local y patrón binario local basado en análisis discriminante lineal para el reconocimiento facial. *Informática Neuronal y Aplicaciones*, 33(13), 7691–7707. <https://doi.org/10.1007/s00521-020-05512-3>
- Navarro, Y., Vera, E., Gonzales, J., & García, J. (2019). *Mediación Tecnológica: Integración de las TIC en investigación, desarrollo y educación*. United Academic Journals.
- Naveen. (2023). *Arquitectura iOS*. Intellipaat Blog. <https://intellipaat.com/blog/tutorial/ios-tutorial/ios-architecture/#:~:text=Architecture%20of%20IOS%20is%20a,of%20well%20defined%20system%20interfaces.>

- Novac, O., Novac, M., Gordan, C., & Berczes, T. (2017). Estudio comparativo de los sistemas operativos móviles Google Android, Apple iOS y Microsoft Windows phone. *14ª Conferencia Internacional Sobre Ingeniería de Sistemas Eléctricos Modernos (EMES)*, 154–159.
- Nunes, A. (2019). *Dispensador modular inteligente para la administración de fármacos*. Universidad de Beira Interior.
- OMS. (2004). *Adherencia a los tratamientos a largo plazo : pruebas para la acción*. Organización Mundial de la Salud.
- Organización Mundial de la Salud. (2003). Incumplimiento del tratamiento prescrito para las enfermedades crónicas. *Comunicados De Prensa*.
- Ortega, J., Sánchez, D., Rodríguez, Ó., & Ortega, J. (2018). Adherencia terapéutica: un problema de atención médica. *Scielo*, 16.
- Pabón, Y., & González, L. (2017). *Formas Farmacéuticas*.  
<https://doi.org/10.16925/greylit.2110>
- Pagès-Puigdemont, N., & Valverde-Merino, M. I. (2018). Adherencia terapéutica: factores modificadores y estrategias de mejora. *Ars Pharmaceutica (Internet)*, 59(4).  
<https://doi.org/10.30827/ars.v59i4.7357>
- Pfizer. (2021). *Pfizer*. <https://www.pfizer.es/tu-salud/Uso-racional-medicamentos/importancia-cumplimiento>
- Quiroz, M., Balseca, A., & Leyva, M. (2021). Estudio cualitativo de reconocimiento de emociones en tiempo real para atención al cliente utilizando Deeplens Face Detection. *Investigación Operacional*, 42(1), 63–73.
- Raspberry Pi. (2020). <https://raspberrypi.cl/producto/camara-v2-para-raspberrypi/>.  
<https://raspberrypi.cl/producto/camara-v2-para-raspberrypi/>
- Raspberry Pi Ltd. (2019). *Raspberry Pi 4 Computer Model B*. [www.raspberrypi.org](http://www.raspberrypi.org)

- Ravi, R., & Yadhukrishna, S. V. (2020, March 1). A Face Expression Recognition Using CNN LBP. *Proceedings of the 4th International Conference on Computing Methodologies and Communication, ICCMC 2020*, 684–689.  
<https://doi.org/10.1109/ICCMC48092.2020.ICCMC-000127>
- Rincón, A. C., Gusñay, N. X., & Rodríguez, V. I. (2020). Therapeutic adherence in patients with chronic diseases of the club of older adults of a health center, Ecuador. *An Real Acad Farm*, 86(2), 125–131. [www.analesranf.com](http://www.analesranf.com)
- Rodriguez, F., Hernandez, A., & Ramirez, J. (2018). Adquisición de datos analógicos con alta precisión usando una Computadora de Placa Única. *RIELAC*, 39, 68–76.  
<https://doi.org/10.1113/jphysiol.1976.sp011363>
- Salazar, J. (2016). Redes inalámbricas. *Techpedia*.  
[https://www.psm.fei.stuba.sk/pages/9/LM01\\_F\\_ES.pdf](https://www.psm.fei.stuba.sk/pages/9/LM01_F_ES.pdf)
- Sarkar, A., Goyal, A., Hicks, D., Sarkar, D., & Hazra, S. (2019). Android application development: a brief overview of Android platforms and evolution of security systems. *Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 73–79. <https://doi.org/10.1109/I-SMAC47947.2019.9032440>
- Terán, G., Cobos, J. C., & Chiluisa, M. (2022). *Microcontroladores* (Universitaria, Ed.; 1st ed.).
- Thomas, P., Delia, L., Corbalan, L., Cáseres, G., Sosa, F., Tesone, F., Cuitiño, A., & Pesado, P. (2018). Tendencias en el desarrollo de Aplicaciones para Dispositivos Móviles. *XX Workshop de Investigadores En Ciencias de La Computación*.
- Toapanta, W. V. C., Palacios, J. del R. G., Chito, A. M. T., & De la Torre, L. A. (2022). Web-mobile application for the management of agricultural producers of the decentralized autonomous government of the Mocha canton. *Universidad y Sociedad*, 14(4).

- Trujillo, M., Aguilar, J., & Neira, C. (2016). Los métodos más característicos del diseño centrado en el usuario -DCU-, adaptados para el desarrollo de productos materiales. *Iconofacto*, 12(19), 215–236. <https://doi.org/10.18566/iconofact.v12.n19.a09>
- Universidad Politécnica de Valencia. (2017). *Arquitectura de Android*.  
<http://www.androidcurso.com/index.php/recursos/31-unidad-1-vision-general-y-entorno-de-desarrollo/99-arquitectura-de-android>
- Untuña, V. (2022). *Sistema de control de acceso por medio de reconocimiento facial con uso de mascarilla y monitoreo de temperatura*. Universidad Técnica de Ambato.
- Valenzuela, E. (2018). *Uso de fármacos en el adulto mayor*.  
<https://medicina.uc.cl/publicacion/uso-farmacos-adulto-mayor/>
- Velasco, J. (2022). *Revisión del Uso del Medicamento (RUM): el servicio más farmacéutico*.  
[https://www.imfarmacias.es/uploads/2022/04/revision\\_medicamento\\_28103\\_20220412023555.pdf](https://www.imfarmacias.es/uploads/2022/04/revision_medicamento_28103_20220412023555.pdf)
- Villanueva, F. (2022). *Metodología de la Investigación* (H. Guerrero, Ed.).

## **Anexos**

### **Anexo 1. Entrevista a especialista en geriatría**

**Nombre:** Dr. Edwin Salazar

**Entidad:** Clínica Metropolitana Ibarra

**Pregunta 1: Cuáles son las enfermedades más comunes en adultos mayores que ha observado en su práctica clínica?**

Respuesta del Dr. Salazar: Bueno, en mi práctica clínica he observado que las enfermedades más comunes en adultos mayores son la diabetes, la hipertensión, el cáncer y, sobre todo, las relacionadas con la demencia, en especial la demencia por Alzheimer. Hay muchas otras



enfermedades que afectan a este grupo, pero estas son las que suelen tener una mayor prevalencia.

**Pregunta 2: Cuáles son los medicamentos que más receta para enfermedades crónicas en adultos mayores, como la diabetes, la hipertensión o la enfermedad cardíaca?**

Respuesta del Dr. Salazar: En general, para enfermedades crónicas como la diabetes, la hipertensión y las enfermedades cardíacas, suelo recetar medicamentos en forma de pastillas o cápsulas. Entre los medicamentos más recetados se encuentran los antihipertensivos, antidiabéticos orales como la metformina e insulina, y para las insuficiencias cardíacas, los betabloqueantes y diuréticos. Hay muchos otros, pero estos son algunos ejemplos comunes.

**Pregunta 3: Cuáles son los medicamentos más comunes que receta para la ansiedad o la depresión en adultos mayores?**

Respuesta del Dr. Salazar: Para tratar la ansiedad y la depresión en adultos mayores, suelo recetar medicamentos como tricíclicos, sertralina y amitriptilina. Estos son ejemplos de fármacos que se utilizan comúnmente para abordar trastornos de salud mental en este grupo de edad.

**Pregunta 4: Cuáles son los medicamentos que se prescriben con mayor frecuencia para tratar enfermedades comunes en los adultos mayores, como las infecciones respiratorias, las infecciones del tracto urinario y las enfermedades gastrointestinales?**

Respuesta del Dr. Salazar: En el caso de enfermedades comunes como infecciones respiratorias, suelo recetar antibióticos como amoxicilina, levofloxacina y claritromicina. Para enfermedades gastrointestinales, algunos de los medicamentos comunes son ciprofloxacina y ampicilina. La elección específica depende del tipo de infección y la respuesta del paciente.

**Pregunta 6: Qué porcentaje de sus pacientes adultos mayores está tomando cinco o más medicamentos al mismo tiempo?**

Respuesta del Dr. Salazar: Cerca del 25% de mis pacientes que permanecen más tiempo en mi consulta consumen cinco o más medicamentos. Sin embargo, en el caso de pacientes atendidos por otros especialistas o de otras instituciones de salud, más del 75% consumen cinco o más medicamentos. Mi objetivo como médico geriatra es reducir el consumo innecesario de medicamentos.

**Pregunta 7: Cuáles cree que son los mayores desafíos que enfrentan los adultos mayores al tomar sus medicamentos correctamente?**

Respuesta del Dr. Salazar: Los mayores desafíos que enfrentan los adultos mayores incluyen la dificultad para recordar cuándo tomar cada medicamento, especialmente aquellos que reciben múltiples recetas de diferentes especialistas. También enfrentan limitaciones físicas o cognitivas para manipular los medicamentos, como pérdida de visión o movilidad. La falta de apoyo o cuidado por parte de un cuidador también puede ser un desafío significativo.

**Pregunta 8: Cree que un dispensador de medicamentos automatizado podría ser útil para los adultos mayores que tienen dificultades para recordar cuándo tomar sus medicamentos?**

Respuesta del Dr. Salazar: Sí, considero que un dispensador de medicamentos automatizado podría ser útil para adultos mayores con dificultades para recordar cuándo tomar sus medicamentos. Sin embargo, es crucial que el diseño se adapte e individualice según las necesidades de cada paciente, teniendo en cuenta limitaciones individuales, como la hipoacusia que podría afectar la eficacia de una alarma auditiva.

**Pregunta 9: Qué características cree que debería tener un dispensador de medicamentos para ser eficaz para los adultos mayores?**

Respuesta del Dr. Salazar: Un dispensador de medicamentos eficaz para adultos mayores debería tener características como facilidad de uso, seguridad en la dispensación de medicamentos, capacidad para almacenar varios medicamentos, flexibilidad en la

programación, alarmas visuales y sonoras, y la opción de programar dosis para varias semanas. Además, el costo debería ser asequible para garantizar su accesibilidad.

## **Anexo 2. Manual de uso del dispensador**

### **Contenido**

1. Introducción
2. Requerimientos básicos
3. Ubicación del dispensador
4. Seguridad
5. Inicio rápido
6. Uso de la aplicación móvil
7. Dispensación de medicamentos
8. Modo viaje: funcionalidad y configuración
9. Mantenimiento y cuidado
10. Problemas comunes y soluciones

### **1. Introducción**

El dispensador de medicamentos es un dispositivo que ayuda a las personas a tomar sus medicamentos de manera segura y cómoda. Está diseñado para pacientes que necesitan tomar medicamentos de forma continua. Este manual proporciona instrucciones detalladas sobre cómo utilizar el dispensador y la aplicación móvil para obtener los mejores resultados.

### **2. Requerimientos básicos**

- **Fuente de alimentación:** Para que el dispensador funcione correctamente, debe estar conectado a una fuente de alimentación eléctrica constante.
- **Conexión a Internet:** Para acceder a todas las funciones y características del dispensador, es necesario que esté conectado a una red wifi activa y estable.

### **3. Ubicación del dispensador**

1. **Ubicar el dispensador en un lugar bien iluminado.** Evitar áreas con sombras o reflejos intensos, ya que pueden afectar a la calidad de las imágenes.
2. **Colocar el dispensador en una superficie plana y estable.** Asegúrese de que el dispensador no esté en riesgo de caerse o volcarse.
3. **Colocar el dispensador a una altura adecuada.** Para que los usuarios puedan alcanzar a la cámara sin ninguna dificultad.

#### 4. Seguridad

Antes de utilizar el dispensador, leer todas las instrucciones y advertencias de seguridad en este manual.

- Mantener en un ambiente limpio y con el compartimiento cerrado.
- Mantener el dispensador fuera del alcance de niños y mascotas.
- Evitar que el dispensador entre en contacto con líquidos o humedad.
- Desconectar el dispensador de la fuente de alimentación antes de limpiarlo.

#### 5. Inicio Rápido

- Conectar el dispensador a la fuente de alimentación utilizando el cable suministrado.
- Conectar el dispensador a una red wifi (en la pantalla se muestra el ícono para que pueda establecer la conexión)

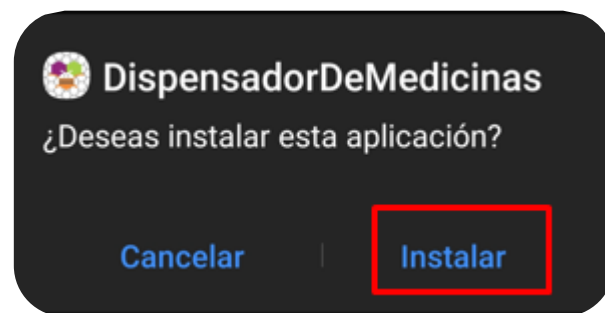
En su teléfono celular:

- Instalar la aplicación móvil con el uso del archivo APK disponible para el sistema operativo Android para esto siga los siguientes pasos:
  1. Descargue el archivo APK que se encuentra en el siguiente enlace [Aplicativo dispensador de medicamentos](#)
  2. Para abrir el archivo que se acaba de descargar, siga estos pasos:
    - Abra la aplicación de Explorador de archivos.
    - Vaya a la carpeta Descargas.

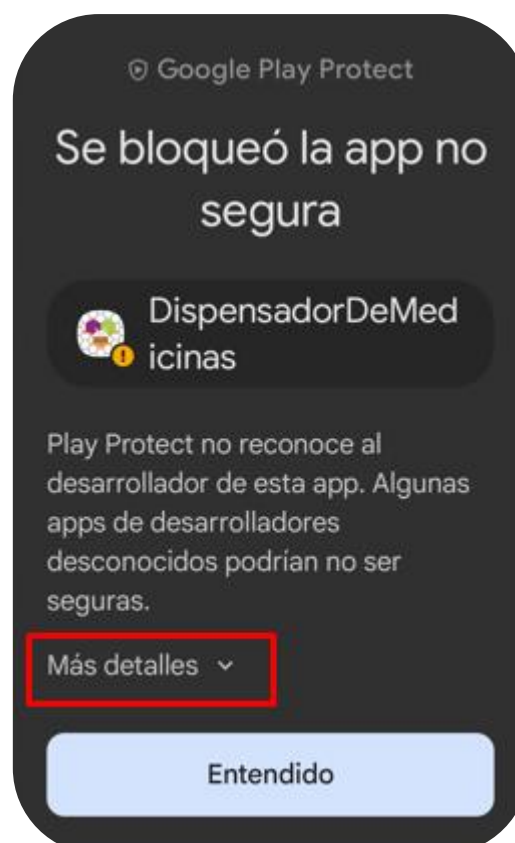
- Toque el archivo que desea abrir



3. Una vez que se haya descargado la aplicación, aparecerá una ventana preguntando si desea instalarla. Presione el botón "Instalar"



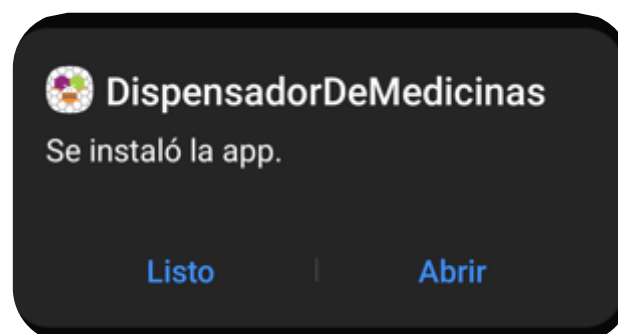
4. Se despliega una nueva ventana. Busque y presione en "Más detalles".



5. posteriormente presione “Instalar de todas formas” y espere hasta que se haya completado la instalación.



6. Al finalizar observará una ventana con el mensaje “Se instaló la app”.



## 6. Uso de la aplicación móvil

Para iniciar sesión en la aplicación, es necesario estar registrado. Si no está registrado, puede crear una cuenta nueva. Para ello, presione el botón "Regístrate" que se encuentra en la pantalla

de inicio de sesión. Este botón abrirá una nueva ventana de registro en la que debe ingresar un nombre de usuario y contraseña. Para este ejemplo, se creó un usuario llamado "usuarioprueba" (no olvide que no debes usar espacios en tu nombre de usuario). Una vez que se haya ingresado el nombre de usuario y la contraseña, presione el botón "Registrar" y ya estará registrado y podrá volver a la pantalla de inicio de sesión.

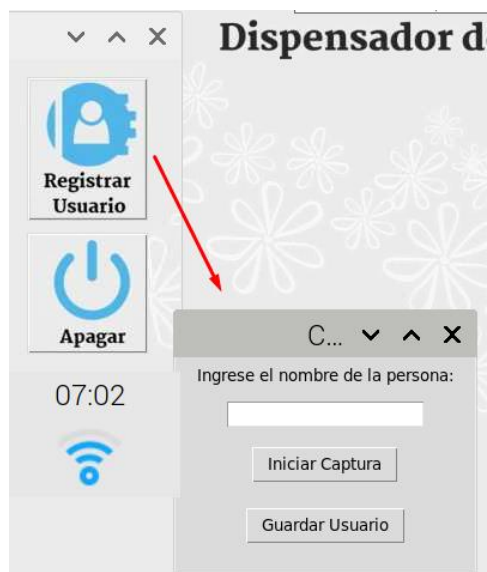


Para iniciar sesión, escriba el nombre de usuario y contraseña que creo anteriormente (para este ejemplo el nombre de usuario es "usuarioapp"). Luego, presione el botón de inicio de sesión. Una vez que haya iniciado sesión, podrá acceder a la pantalla de registro de medicamentos.



- **Registro de rostro para reconocimiento facial**

Como primer paso para el registro es necesario encender el dispensador, donde se podrá visualizar 3 botones y para registrar deberemos presionar el primer botón que dice registro de usuario, y se nos desplegará una ventana donde debemos de presionar en la casilla en blanco.

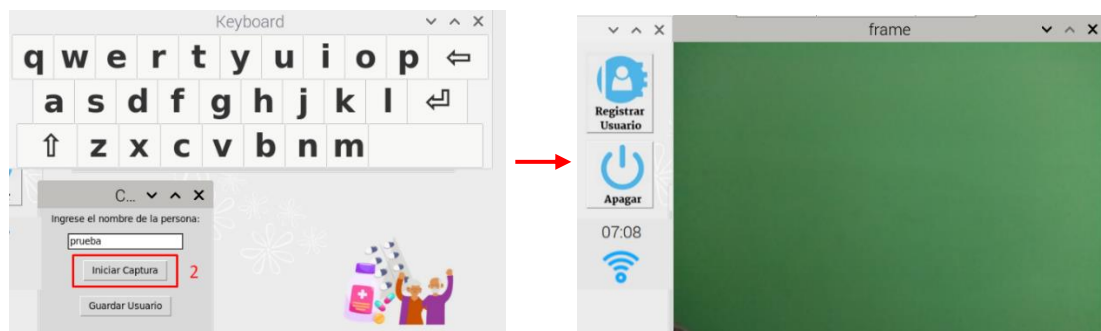


Una vez presionado, se nos desplegará un teclado donde escribiremos el nombre de usuario de la persona que se registró en la aplicación anteriormente.

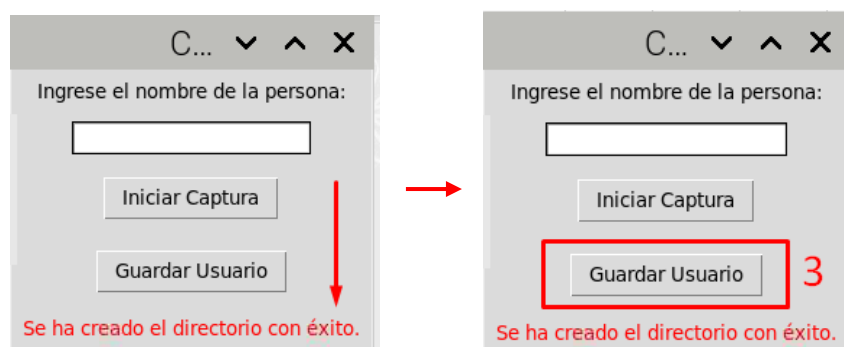




Seleccionaremos el botón de iniciar captura para que se nos despliegue una ventana donde se visualizara la cámara y colocaremos el rostro del usuario.



Una vez hecho eso se observará un mensaje donde nos informara que se ha creado correctamente el directorio, por último, deberemos colocar en guardar usuario para que el algoritmo de reconocimiento empiece el entrenamiento con el nuevo usuario.



- **Programación de horarios para la toma de medicación**

Una vez que se despliega la pantalla de registro de alarmas, puede configurar cada uno de los compartimientos. Para esto, primero agregue el nombre en el campo correspondiente. En este ejemplo, el nombre del medicamento es “Losartan”.



Luego, seleccione el número del compartimiento que desee utilizar en la lista desplegable. Los compartimientos están numerados del 1 al 6. En este ejemplo, seleccionamos el compartimiento 1.



Ahora debe especificar el número de pastillas que ha ingresado en el compartimiento seleccionado. A través de la lista desplegable, selecciona el número de pastillas. Por ejemplo, si ha ingresado 10 pastillas en el compartimiento, seleccione el número 10 en la lista desplegable.



Para configurar la hora de la alarma, siga estos pasos:

1. En la pantalla, verá tres listas desplegables.
2. En la primera lista, seleccione la hora.
3. En la segunda lista, seleccione los minutos.
4. En la tercera lista, seleccione la notación del horario (am o pm).

Por ejemplo, si quiere programar la alarma para la toma de un medicamento a las 7:00 a.m., seleccionaría la hora 7 en la primera lista, los minutos 00 en la segunda lista y la notación am en la tercera lista.



Una vez que haya completado todos los campos, presione el botón registrar/modificar para que todos los cambios se guarden correctamente. Repita el proceso para cada uno de los compartimientos.



- **Interfaz del dispensador**

Al momento de encender el dispensador se indicara la siguiente ventana donde se tiene el estado del dispensador con el historial en tiempo real de los medicamentos y su posición en el

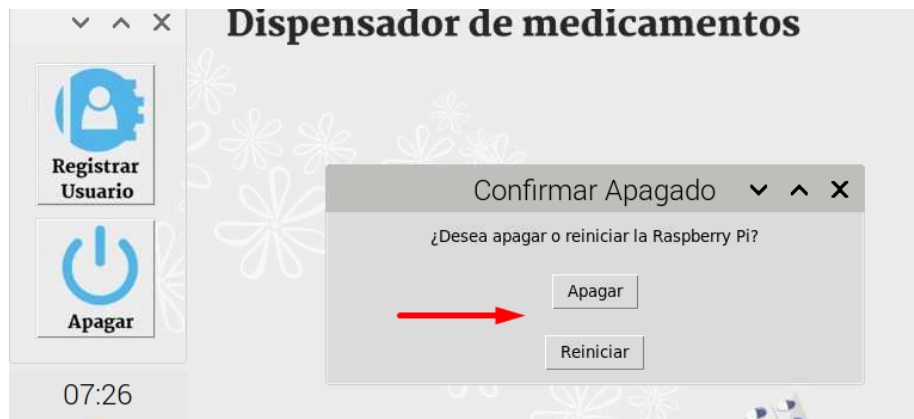
dispensador, también consta con los botones que nos sirve tanto para registro, apagado del sistema y configuración WiFi.



Para la configuración WiFi se debe de presionar el ícono y nos saltara la ventana como se muestra a continuación donde seleccionaremos la red que deseemos y colocaremos la contraseña correctamente.



Para el botón de apagado basta con presionarlo y se desplegará la siguiente ventana donde debemos de colocar lo deseado ya sea apagar el sistema o bien reiniciarlo.

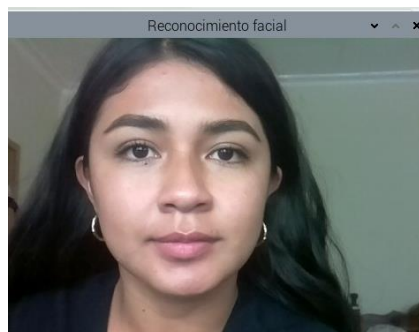


## 7. Dispensación de medicamentos

Una vez que se cumpla el horario programado mediante la aplicación, el bot de Telegram enviará una alerta a su dispositivo celular. La alerta incluirá el número del compartimiento del cual se dispensará la pastilla.

Para dispensar medicamentos

1. Acérquese al dispensador.
2. Cuando se despliegue la ventana donde se podrá observar la cámara y deberá colocarse al frente del dispositivo para que funcione el reconocimiento facial.
3. Colóquese frente al dispensador de tal forma que su rostro quede completamente visible para la cámara (deberá ver su rostro completamente en la pantalla) como se muestra en la siguiente figura y manténgase hasta que dicha pantalla desaparezca.



4. El dispensador reconocerá su rostro y abrirá el compartimiento correspondiente al medicamento programado.

5. Retire su medicamento de la bandeja ubicada en la parte lateral izquierda del dispensador.
6. Introduzca la bandeja en la ranura correspondiente.

### 8. Modo viaje: funcionalidad.

Para utilizar el modo viaje, diríjase a la ventana registro de alarmas, presione el botón “modo viaje” y se desplegará una nueva pantalla. Para dispensador el medicamento, es necesario que



Para acceder a la funcionalidades del modo viaje, el usuario unicamente debe acceder a la pantalla modo viaje y presionar expulsar en el compartimiento en el que se encuentre su medicamento para que este pueda ser expulsado



## 9. Mantenimiento y cuidado

- Limpie la pantalla y la cámara con un paño suave y limpio regularmente para eliminar la suciedad, el polvo y las huellas dactilares.
- Mantenga el dispensador enchufado para evitar interrupciones en la administración de medicamentos.

## 10. Problemas comunes y soluciones

- Si el dispensador no se enciende, verifique la conexión de la fuente de alimentación.
- Si el reconocimiento facial no funciona, asegúrese de estar bien iluminado y sin obstáculos frente a la cámara.

## Anexo 3. Códigos implementados en la Raspberry Pi

Enlace a los códigos implementados en la Raspberry Pi

<https://github.com/GeovanaColimba/Dispensador-de-medicamentos>