

**UNIVERSIDAD DE TARAPACÁ**  
*Universidad del Estado*



Departamento de Ingeniería en Computación e Informática



# **Plan de proyecto**

## **Control de asistencia biométrica**

**Autor(es): Sebastián Cayupi**

**Manuel Lopez**

**Juan Meneses**

**Asignatura: Proyecto II**

**Profesor: Diego**

**Aracena**

## Historial de Cambios

| Fecha      | Versión | Descripción                                       | Autor(es)  |
|------------|---------|---|--|
| 10/09/2024 | 1.0     | Versión preliminar del formato                    | Sebastián Cayupi<br>Manuel Lopez                 |
| 10/09/2024 | 1.0.1   | Se arregló errores denotados por el profesor      | Sebastián Cayupi<br>Manuel Lopez                 |
| 24/09/2024 | 1.1     | Se realizó la planificación de gestión de riesgos | Sebastián Cayupi<br>Manuel Lopez<br>Juan Meneses |
| 05/11/2024 | 2.0     | Se actualizó con los requerimientos de la Fase II | Sebastián Cayupi<br>Manuel Lopez<br>Juan Meneses |
|            |         |   |  |

# 1. Tabla de contenidos

|  |    |
|--|----|
| 1. Tabla de contenidos                         | 3  |
| 2. Panorama General                            | 4  |
| 2.1. Resumen del proyecto                      | 4  |
| 2.1.1 Objetivo general                         | 4  |
| 2.1.2. Objetivos específicos                   | 4  |
| 2.2.3. Suposiciones y Restricciones            | 5  |
| 2.2.4. Entregables del proyecto                | 5  |
| 3. Organización del proyecto                   | 6  |
| 3.1. Personal y entidades internas             | 6  |
| 3.2. Roles y Responsabilidades                 | 6  |
| 3.3. Mecanismos de comunicación y organización | 7  |
| 4. Planificación de los procesos de gestión    | 7  |
| 4.1. Planificación de recursos                 | 7  |
| 4.1.1. Programación del proyecto               | 7  |
| 4.1.2. Electrónica del proyecto                | 9  |
| 4.1.3. Recursos Humanos                        | 12 |
| 4.2. Costos totales                            | 14 |
| 4.2.1. Recursos tangibles                      | 14 |
| 4.2.2. Recursos intangibles                    | 14 |
| 4.2.3. Tabla de costos totales                 | 14 |
| 4.3. Distribución de tiempos                   | 15 |
| 4.3.1. Asignación de tiempos                   | 16 |
| 4.4. Planificación de la gestión de riesgos    | 16 |
| 5. Planificación de los procesos técnicos      | 18 |
| 5.1. Modelos de proceso                        | 18 |
| 5.1.1. Requerimientos                          | 18 |
| 5.1.2. Casos de uso                            | 20 |
| 5.1.3. Interfaz de usuario                     | 34 |
| Bocetos de Pantallas Principales               | 34 |
| 5.2. Herramientas y técnicas                   | 36 |
| 6. Conclusión                                  | 37 |
| 7. Referencias                                 | 38 |

Falta indice figuras y tablas

## 2. Panorama General

### 2.1. Resumen del proyecto

El control de asistencia en las instituciones educativas sigue siendo un proceso que a menudo enfrenta problemas de precisión, eficiencia y carga administrativa. En la Universidad de Tarapacá, estos retos pueden afectar el enfoque de los docentes en la enseñanza y la gestión del tiempo en el aula. Este proyecto propone la implementación de un sistema de control de asistencia automatizado que utiliza tecnología biométrica y captura de imágenes para optimizar y automatizar el proceso de registro de asistencia, mejorando la precisión y reduciendo los errores humanos.

El proyecto "Control de Asistencia Biométrico" busca simplificar y hacer más eficiente la toma de asistencia, permitiendo que los profesores dediquen más tiempo a sus actividades académicas y reduciendo la carga administrativa. Además, ofrece una mayor fiabilidad al proceso de registro y ayuda a evitar posibles manipulaciones o errores en la asistencia manual.



#### 2.1.1 Objetivo general

El objetivo de este proyecto es el siguiente:

Desarrollar un sistema automatizado de control de asistencia basado en tecnología biométrica que permita registrar de manera precisa y eficiente la asistencia de los estudiantes, disminuyendo la carga administrativa de los profesores y aumentando la seguridad en el proceso de registro.



#### 2.1.2. Objetivos específicos

- Investigar y definir los recursos necesarios para la implementación del sistema.
- Diseñar el sistema de control de asistencia biométrica, asegurando su integración con la red de la universidad.
- Planificar las fases de desarrollo y pruebas del proyecto.
- Implementar el software y los dispositivos (Raspberry Pi, sensores de huellas dactilares, cámaras) para garantizar la funcionalidad del sistema.
- Realizar pruebas piloto del sistema para verificar su funcionamiento y hacer ajustes en base a los resultados obtenidos.



### 2.2.3. Suposiciones y Restricciones

Se espera que el proyecto sea un éxito se deberá cumplir con las siguientes cuestiones:

- Los estudiantes tendrán la capacidad y disposición para utilizar el sistema de control de asistencia biométrica sin inconvenientes.
- Los profesores estarán dispuestos a integrar el uso del sistema en sus clases y a adaptarse al nuevo método de registro de asistencia.
- Los recursos necesarios para construir e implementar el sistema, como sensores biométricos, Raspberry Pi y cámaras, estarán disponibles a tiempo. ✓
- El equipo de proyecto cuenta con la experiencia y habilidades técnicas suficientes para desarrollar e implementar el sistema correctamente.



separar

- **Las restricciones para este proyecto son las siguientes:**
- **Compatibilidad entre software y hardware:** Versiones compatibles exactamente con otra.
- **Económico:** El costo de los elementos (sensores, cables, entre otros) debe ser de acuerdo a un presupuesto.
- **Usuarios o público definido:** El sistema está destinado a ser utilizado únicamente por estudiantes y profesores de la Universidad de Tarapacá.

### 2.2.4. Entregables del proyecto

Los entregables del proyecto son los siguientes:

1. Primera Presentación del proyecto
2. Maqueta del sistema
3. Informe del proyecto (primera parte)
4. Segunda Presentación del proyecto
5. Informe del proyecto (segunda parte)
6. Redmine uta (Wiki, bitácoras y Carta Gantt) ✓
7. Manual de usuario
8. Sistema "Sensor Biométrico"

## 3. Organización del proyecto

### 3.1. Personal y entidades internas

| Entidades Internas | Personal encargado | Descripción  |
|--------------------|--------------------|--|
| Jefe de Proyecto   | Sebastian Cayupi   | El jefe de proyecto debe ser la persona encargada de gestionar y organizar el trabajo en equipo como también asegurarse de que las tareas se dividan de manera equitativa.                               |
| Jefe Analista      | Manuel Lopez       | Un jefe analista tiene como objetivo principal el diseño y planificación del proyecto. Este se enfoca en comprender e investigar acerca de los requerimientos y restricciones relacionadas al "Sistema". |
| Jefe de Tecnología | Juan Meneses       | Un jefe de tecnología es quien investiga los artefactos tecnológicos a utilizar, es decir, se encarga de su implementación, mantenimiento y desarrollo.  |

### 3.2. Roles y Responsabilidades



Los roles que tomarán cada integrante del equipo serán:

|                                      |  |
|--------------------------------------|--|
| <b>Programador</b>                   | Se encargará de diseñar la arquitectura de software, desarrollar aplicaciones y sistemas de control, además de realizar pruebas de funcionamiento del software para su depuración.   |
| <b>Técnico de hardware</b>           | Encargado de seleccionar los componentes de hardware (como sensores y actuadores) y ensamblar los prototipos de hardware, desarrollar pruebas del mismo y solucionar los problemas que se presenten relacionados con el funcionamiento del hardware. |
| <b>Diseñador</b>                     | Encargado de diseñar presentaciones, diagramas e interfaces y transmitir información compleja a través de ellas. Se encarga también de asegurar un mínimo de calidad y atractividad para el material audiovisual que genera.                         |
| <b>Técnico de redes</b>              | Encargado de investigar sobre las herramientas y conocimientos necesarios para poder implementar una red y diseñar su estructura.  |
| <b>Especialista en documentación</b> | Mantiene la documentación del proyecto actualizada manteniendo un registro constante de la información relevante para un proyecto.   |

### 3.3. Mecanismos de comunicación y organización

Un proyecto necesita coordinación entre los integrantes del equipo, es por ello que para el desarrollo del sistema del Sensor Biométrico se requieren de los siguientes medios de comunicación

- **Whatsapp:** Será el principal medio de comunicación por el cual los integrantes poseen mayor disponibilidad.
- **Discord:** Utilizado para reuniones, discord es un excelente medio para las reuniones y presentación en el equipo.
- **Google Drive:** Organización de archivos, carpetas y avances del sistema, google drive nos brindará un espacio en donde se pueda almacenar la información relacionada.
- **Redmine:** Ideal para almacenar la documentación y el seguimiento de los plazos.

## 4. Planificación de los procesos de gestión

### 4.1. Planificación de recursos

#### 4.1.1. Programación del proyecto

Dentro de los proyectos relacionados con el Internet de las Cosas (IoT), es indiscutible la centralidad y relevancia que posee el software. Si bien los dispositivos electrónicos y los sensores son el aspecto exterior de cualquier sistema, es el software el que les da vida, inteligencia y la capacidad de trabajar en conjunto de manera programada y lógica.

En esta sección, se analizarán los aspectos financieros esenciales para la implementación del software de este proyecto.

#### Equipos para la programación

Para el apartado de programación del código necesario para el proyecto, se dispondrán equipos (notebooks) personales para cada integrante del grupo. Total notebooks: 3

## Software externo utilizado

## De uso libre

| Nombre del Software | Uso dado   | Condiciones de uso  |
|---------------------|--|---|
| Visual Studio Code  | Utilizado para programar los programas necesarios  | Libre uso bajo los <a href="#">Términos de licencia de Microsoft</a>      |
| Lucidchart          | Utilizado para diagramar y esquematizar interacciones con el sistema (usando UML)        | Plan gratuito bajo los <a href="#">Términos de uso del Software Lucid</a> |
| Canva               | Utilizado para diseñar diagramas demostrativos   | Libre uso bajo los <a href="#">Términos de uso de Canva</a>               |
| WhatsApp            | Utilizado para la comunicación informal del grupo de trabajo                             | Libre uso bajo los <a href="#">Términos de servicio de WhatsApp</a>       |
| Discord             | Utilizado para la realización de reuniones remotas del grupo de trabajo                  | Libre uso bajo los <a href="#">Términos de servicio de Discord</a>        |
| Redmine             | Utilizado para planear y organizar tareas y documentos del proyecto                      | Proporcionado por la institución UTA                                      |
| Unity 3D            | Utilizado para la confección de la maqueta 3D y su ejemplificación de la idea a realizar | Software de libre disposición. Véase <a href="#">Términos de licencia</a> |

## Pagados

| Nombre del Software | Uso dado   | Coste de uso                                    |
|---------------------|--|---|
| Microsoft 365       | Utilizado para suplir necesidades de ofimática (documentos y presentaciones) | \$34.999 CLP/mes<br>(2-6 personas por licencia) |



## 4.1.2. Electrónica del proyecto

Además del software, el hardware es fundamental para asegurar el funcionamiento del sistema de control de asistencia biométrica. A continuación, se detallan los componentes electrónicos que se usarán en este proyecto.

### Raspberry Pi 4

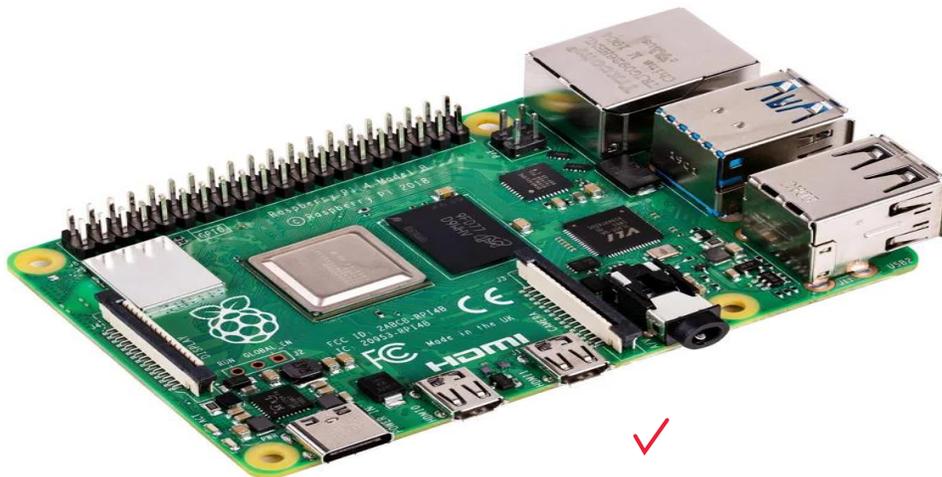
Es el componente central del sistema, actuando como el procesador principal que conecta y gestiona las comunicaciones entre los sensores biométricos y el software de control de asistencia. La Raspberry Pi 4 contiene el código necesario para la operación del sistema y facilita la interacción entre el usuario y el proceso de registro.

El proyecto utilizará el sensor biométrico **Adafruit Fingerprint**, que es esencial para la captura de huellas dactilares de los estudiantes al momento de ingresar al aula. Este sensor se integra fácilmente con la Raspberry Pi y permite un registro preciso y seguro de las huellas digitales.

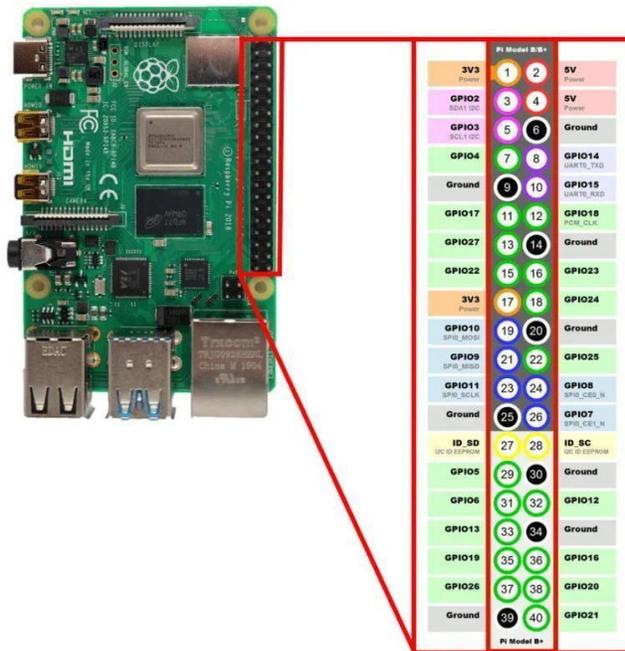
#### Características del sensor Adafruit Fingerprint:

**Precisión y fiabilidad:** Este sensor es conocido por su capacidad de reconocer huellas dactilares de forma rápida y precisa.

**Interfaz sencilla:** Se conecta a través de interfaces UART, lo que facilita su comunicación con la Raspberry Pi.



## Especificaciones técnicas



- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- 1GB, 2GB or 4GB LPDDR4-2400 SDRAM (depende del modelo)
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 USB 3.0 ports; 2 USB 2.0 ports.
- GPIO de 40 pines estándar de Raspberry Pi (tiene total retrocompatibilidad con modelos anteriores)
- 2 puertos micro-HDMI (soporta hasta 4k 60fps)
- 2-lane MIPI DSI display port
- 2-lane MIPI CSI camera port
- Puerto de 4 polos de audio estéreo y vídeo componente
- H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode)
- OpenGL ES 3.0 graphics
- Puerto Micro-SD para el sistema operativo y almacenamiento masivo
- 5V DC por conector USB-C (mínimo 3A\*)
- 5V DC por header GPIO (mínimo 3A\*)
- Power over Ethernet (PoE) enabled (necesita un PoE HAT aparte)
- Temperatura de trabajo: 0 – 50 grados C ambiente

*Se puede usar un cargador de buena calidad de 2.5A si la carga total de los puertos USB es inferior a 500mA*



## Sensores

Complementando los objetivos principales del proyecto, la finalidad que se tiene con los sensores es la manipulación de datos biométricos, en este caso por medio de las huellas digitales.

Lista de sensores:

1. **Considerados**

- a. Sensor de huella digital
- b. Sensor de cámara
- c. pantalla



| Sensor                   | Función                      | Descripción  |
|--------------------------|------------------------------|--|
| Sensor de huella digital | Identificación de usuarios   | Permite identificar a los usuarios mediante sus huellas dactilares. Brinda seguridad y precisión en la toma de asistencia. |
| Sensor de cámara         | Supervisión de entorno       | Proporciona una imagen visual del entorno para verificar asistencia y evitar suplantación de identidad.                    |
| Pantalla                 | Visualización de información | Muestra mensajes de confirmación de asistencia, errores, o el estado del sistema para el usuario.                          |



### 4.1.3. Recursos Humanos

Para el desarrollo del proyecto se consideran 16 semanas de trabajo continuo. Las horas de trabajo de cada rol se organizan y costean de la siguiente manera:

#### Costos hora por rol

| Rol                           | Costo/hora       | Personal que cumple el rol                   |
|-------------------------------|------------------|--|
| Programador                   | \$5.292 CLP/hora | Sebastián Cayupi, Manuel Lopez, Juan Meneses |
| Técnico de hardware           | \$3.077 CLP/hora | Sebastián Cayupi, Manuel Lopez               |
| Especialista en documentación | \$2.769 CLP/hora | Juan Meneses                                 |

#### Tiempo dedicado a cada rol (por tarea específica)

Las horas semanales mostradas a continuación no necesariamente se cumplen al mismo tiempo. Se considera la totalidad del proyecto al hablar de semanas y horas semanales.

| Tarea específica       | Semanas | Horas Semanales | Horas Totales |
|------------------------|---------|-----------------|---------------|
| Programación           | 9 sem.  | 6 hrs./sem.     | 54 hrs.       |
| Preparación Hardware   | 11 sem. | 3 hrs./sem.     | 33 hrs.       |
| Documentar el proyecto | 16 sem. | 1.5 hrs./sem.   | 24 hrs.       |
| <b>Total horas</b>     |         | 10.5 hrs./sem.  | 111 hrs.      |

## Costo de la contratación

| Rol contratado                      | Horas requeridas p./p. | N° personas encargadas | Total horas requeridas | Costo total contratación |
|-------------------------------------|------------------------|------------------------|------------------------|--------------------------|
| Programador                         | 54 hrs.                | 3                      | 162 hrs.               | \$847.422 CLP            |
| Técnico de hardware                 | 33 hrs.                | 2                      | 66 hrs.                | \$203.082 CLP            |
| Especialista en documentación       | 24 hrs.                | 3                      | 72 hrs.                | \$398.736 CLP            |
| <b>Costo total recursos humanos</b> |                        |                        |                        | <b>\$1.449.240 CLP</b>   |



## 4.2. Costos totales

### 4.2.1. Recursos tangibles

| Elemento               |                          | Uso  | Unidades     | Tipo de pago | Coste unitario est. | Coste total est. |
|------------------------|--------------------------|--|--------------|--------------|---------------------|------------------|
| Notebooks              |                          | Equipos dedicados a la codificación                          | 3            | Un solo pago | \$500.000 CLP       | \$1.500.000 CLP  |
| Raspberry Pi (4GB RAM) |                          | Ser la pieza principal del sistema                           | 1            | Un solo pago | \$79.000 CLP        | \$79.000 CLP     |
| Sensores               | Sensor de huella digital | Detectar, medir y digitalizar informaciones del mundo físico | 1 (p/sensor) | Un solo pago | ~ \$20.000 CLP      | ~ \$45.000 CLP   |
|                        | Sensor de cámara         |  |              |              | ~ \$5.000 CLP       |                  |
|                        | Pantalla                 |  |              |              | ~ \$20.000 CLP      |                  |

### 4.2.2. Recursos intangibles

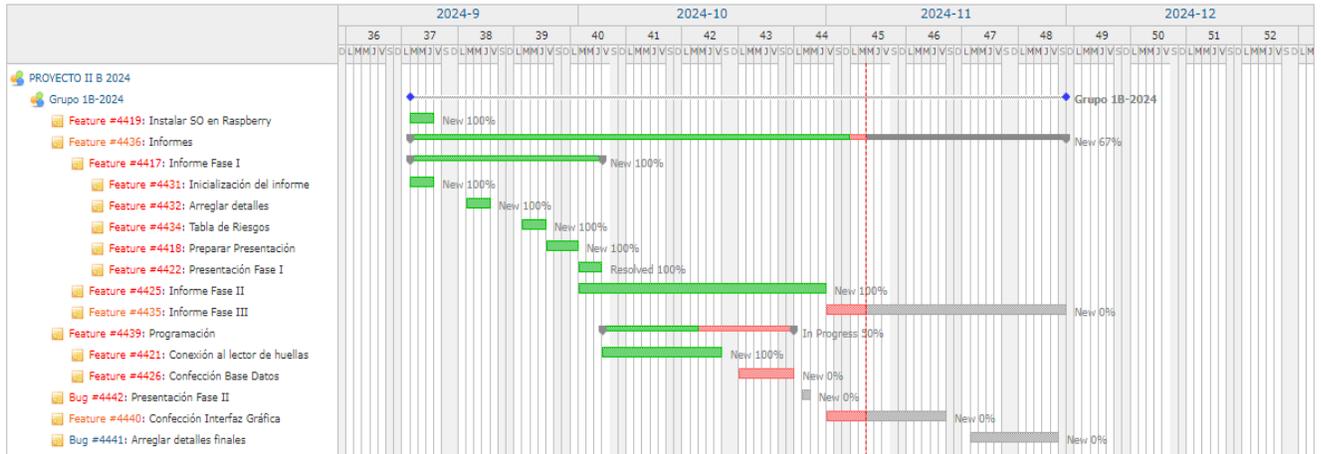
| Elemento                     | Uso  | Unidades                     | Tipo de pago | Coste unitario est.       | Coste total est. |
|------------------------------|--|------------------------------|--------------|---------------------------|------------------|
| Software (Licencias pagadas) | Facilitar tareas administrativas y de desarrollo | 1 (servicio por suscripción) | Mensual      | \$6.990 CLP/mes (5 meses) | \$34.950 CLP     |

### 4.2.3. Tabla de costos totales

| Elemento                    | Costo                  |
|-----------------------------|------------------------|
| Recursos tangibles          | \$1.624.000 CLP        |
| Recursos intangibles        | \$34.950 CLP           |
| Recursos humanos            | \$1.449.240 CLP        |
| <b>Costo total proyecto</b> | <b>\$3.108.190 CLP</b> |

### 4.3. Distribución de tiempos

Se ha logrado de manera parcial los objetivos planteados con la carta gantt inicial, unos pequeños retrasos pero todo bastante realizable dentro de los tiempos.



falta casi la mitad de la gantt

### 4.3.1. Asignación de tiempos

Para el desarrollo del proyecto se contemplan 16 semanas de trabajo, para las cuales, se trabajará una media de 12 horas semanales.

La duración total del proyecto está estimada en 192 horas de trabajo repartidas en 4 meses de desarrollo.



### 4.4. Planificación de la gestión de riesgos

Los posibles riesgos que se puede enfrentar el equipo se encuentran en la siguiente tabla

| #  | Riesgo   | Probabilidad | Impacto         |
|----|--|--------------|-----------------|
| 1  | Personal sin experiencia.                                    | 60%          | Crítico(2)      |
| 2  | Mala planificación de tiempo                                 | 30%          | Marginal(3)     |
| 3  | Falta de participación de integrantes.                       | 10%          | Crítico(2)      |
| 4  | Falta de componentes o problemas con el hardware.            | 25%          | Crítico(2)      |
| 5  | Abandono de proyecto por integrantes.                        | 10%          | Crítico(2)      |
| 6  | Retraso en el reconocimiento de huellas.                     | 50%          | Crítico(2)      |
| 7  | Pérdida de información.                                      | 5%           | Catastrófico(1) |
| 8  | Pérdida de energía.  | 10%          | Crítico(2)      |
| 9  | Fallo en la conectividad de red.                             | 10%          | Crítico(2)      |
| 10 | Incompatibilidad de hardware o software.                     | 15%          | Marginal(3)     |
| 11 | Fallo al identificar huellas sucias, en mal estado, húmedas. | 20%          | Marginal(3)     |



Las acciones para remediar estas situaciones son las siguientes:

1. **Personal sin experiencia:** Estudiar componentes a usar y practicar su aplicación según el proyecto.
2. **Mala planificación de tiempo:** Identificar rol o roles que no cumpla con fecha de entrega y apoyar al rol.
3. **Falta de participación de integrantes:** Asignar trabajo particular al integrante y conversar la situación.
4. **Falta de componentes o problemas con el hardware:** Comprar componentes con anticipación y tener piezas de repuesto.
5. **Abandono de proyecto por integrantes:** Redistribuir las tareas entre el equipo restante e intentar adelantar lo posible
6. **Retraso en el reconocimiento de huellas:** Optimizar el código de reconocimiento y ajustar tiempos de procesamiento
7. **Pérdida de información:** Implementar backups automáticos y usar sistemas de control de versiones.
8. **Pérdida de energía:** Implementar una batería de respaldo para la Raspberry Pi.
9. **Fallo en la conectividad de red:** Verificar la red y el cableado.
10. **Incompatibilidad de hardware o software:** Verificar la compatibilidad antes de comprar componentes y hacer pruebas tempranas de funcionalidad.
11. **Fallo al identificar huellas sucias, en mal estado, húmedas.:** Mantener las recomendaciones para usar de óptima forma el sensor, pegar en la caja (secar siempre las manos antes de usar, limpiarse los dedos con agua, alcohol)



## 5. Planificación de los procesos técnicos

### 5.1. Modelos de proceso

#### 5.1.1. Requerimientos

Los requerimientos funcionales y no funcionales son pilares fundamentales en el diseño y desarrollo de sistemas, proporcionando la estructura esencial para la creación de soluciones tecnológicas que cumplen con las necesidades y expectativas de sus usuarios.

#### Requerimientos funcionales

A continuación ~~presentamos~~ los requerimientos funcionales de nuestro sistema.

#### Requerimientos funcionales:

- El sistema debe capturar y registrar la asistencia de los estudiantes mediante el uso de un sensor de huellas dactilares.
- El software debe procesar y enviar los datos a la intranet de la universidad para mantener un registro actualizado.
- El sistema debe incluir una función de captura de imágenes para respaldar la autenticidad del registro de asistencia.
- Debe contar con una interfaz de usuario amigable para la gestión de registros por parte de los docentes.
- Debe permitir la revisión y exportación de los registros de asistencia en formato digital.

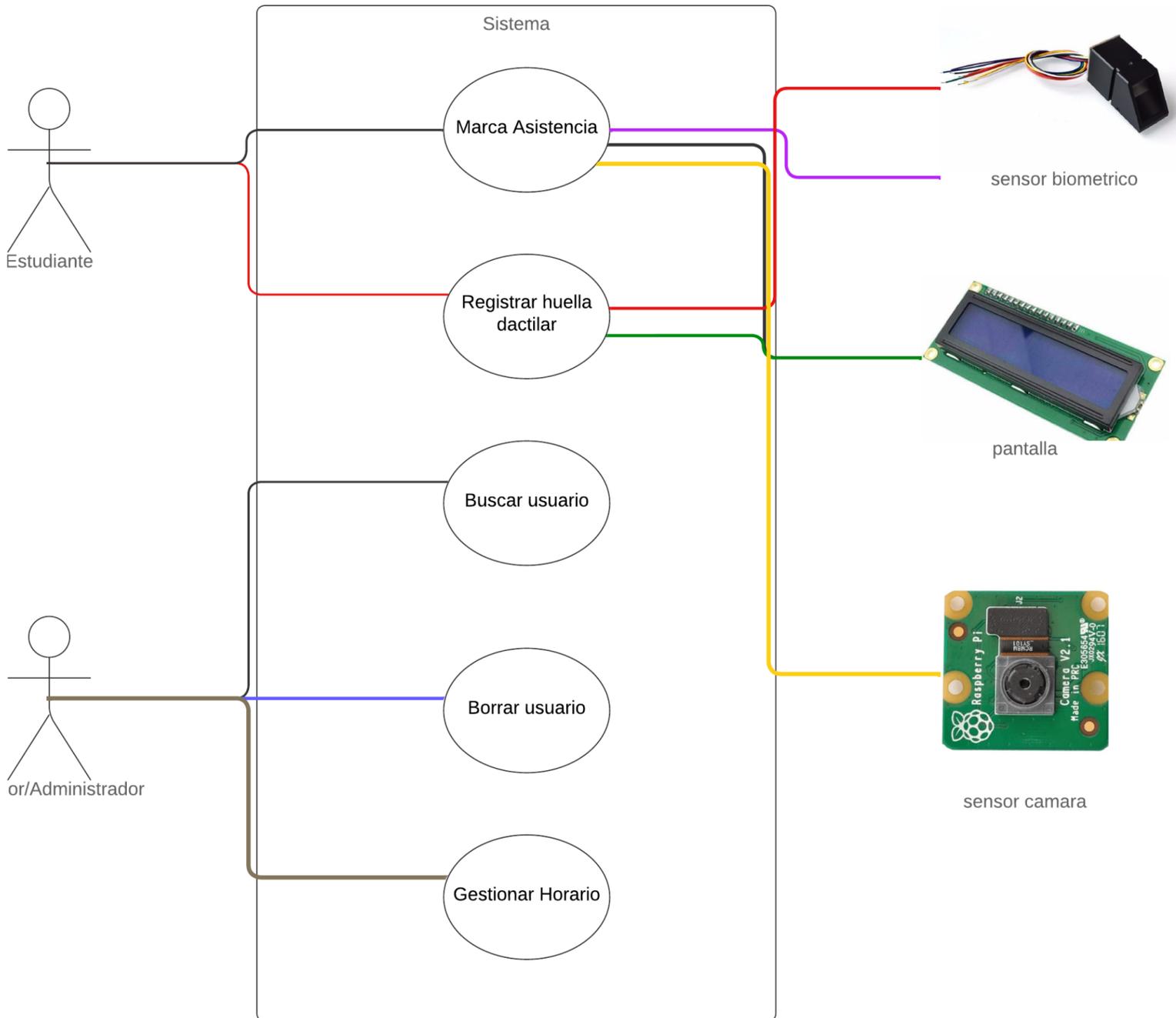


## Requerimientos no funcionales

- **Seguridad:** Los datos biométricos deben ser almacenados y transmitidos de forma cifrada para proteger la privacidad de los usuarios.
  - **Usabilidad:** La interfaz debe ser intuitiva y fácil de usar tanto para estudiantes como para profesores.
  - **Escalabilidad:** El sistema debe ser capaz de integrarse y adaptarse a nuevas funcionalidades o dispositivos si se requiere en el futuro.
  - **Rendimiento:** El tiempo de respuesta del sistema debe ser inferior a 2 segundos por registro de asistencia.
  - **Compatibilidad:** Debe ser compatible con la infraestructura tecnológica existente de la universidad, como la red y las bases de datos.
- 

### 5.1.2. Casos de uso

Diagrama de casos de uso general



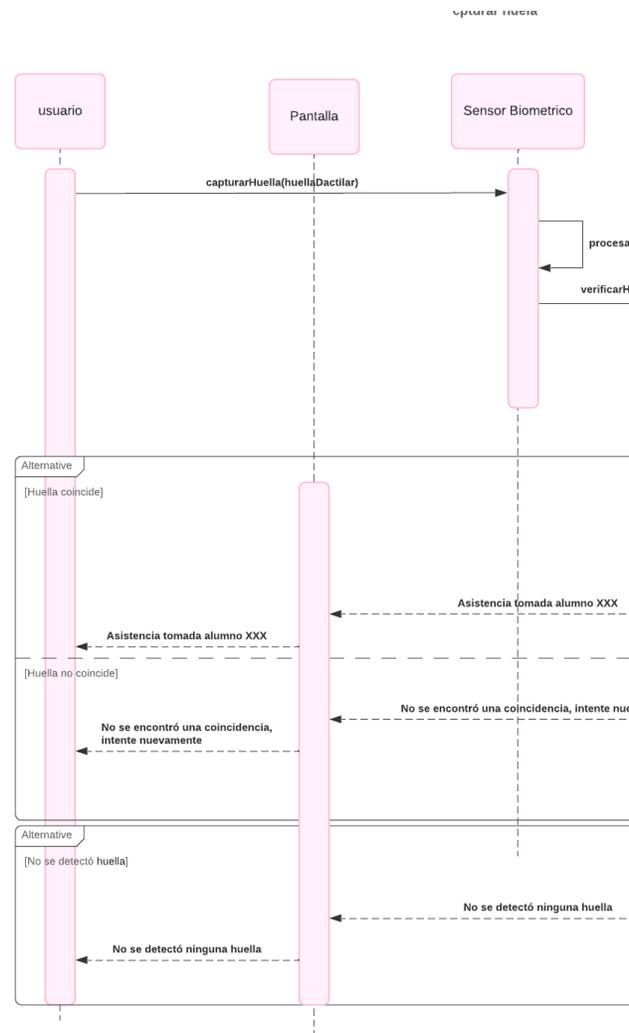
(caso de uso general img Uso 01)



Descripción de casos de uso.

|   |   |
|---|---|
| Nombre cus: Marcar Asistencia   |   |
| Descripción: Permite registrar la huella dactilar de un estudiante en el sistema, utilizando un sensor biométrico, para que pueda ser identificado en el sistema de asistencia.                         |   |
| Actor: Estudiante, profesor/administrador, sensor biométrico, raspberry pi, pantalla, camara sensor   |   |
| Precondición: <ol style="list-style-type: none"> <li>1. El estudiante debe ser alumno regular del sistema.</li> <li>2. El alumno debe contar con una huella digital ingresada en el sistema.</li> </ol> |   |
| Flujo Principal: Actor <ol style="list-style-type: none"> <li>1. Ingresar su huella en el sensor.</li> </ol>  | Flujo Principal: Sistema <ol style="list-style-type: none"> <li>2. se muestra en pantalla "asistencia tomada alumno XXX"</li> </ol>                       |
| Flujo Alternativo:  | <ol style="list-style-type: none"> <li>2.1. Se muestra un mensaje en pantalla: "No se detectó ningún dedo, por favor intente nuevamente"</li> </ol>       |
|   | <ol style="list-style-type: none"> <li>2.2. muestra un mensaje en pantalla : " No se encontró una coincidencia, por favor intente nuevamente."</li> </ol> |
| Postcondiciones:La asistencia del usuario queda registrada en el sistema correctamente.   |   |
| Valor medible:Reduce el tiempo usado para tomar la asistencia en clases.  |   |



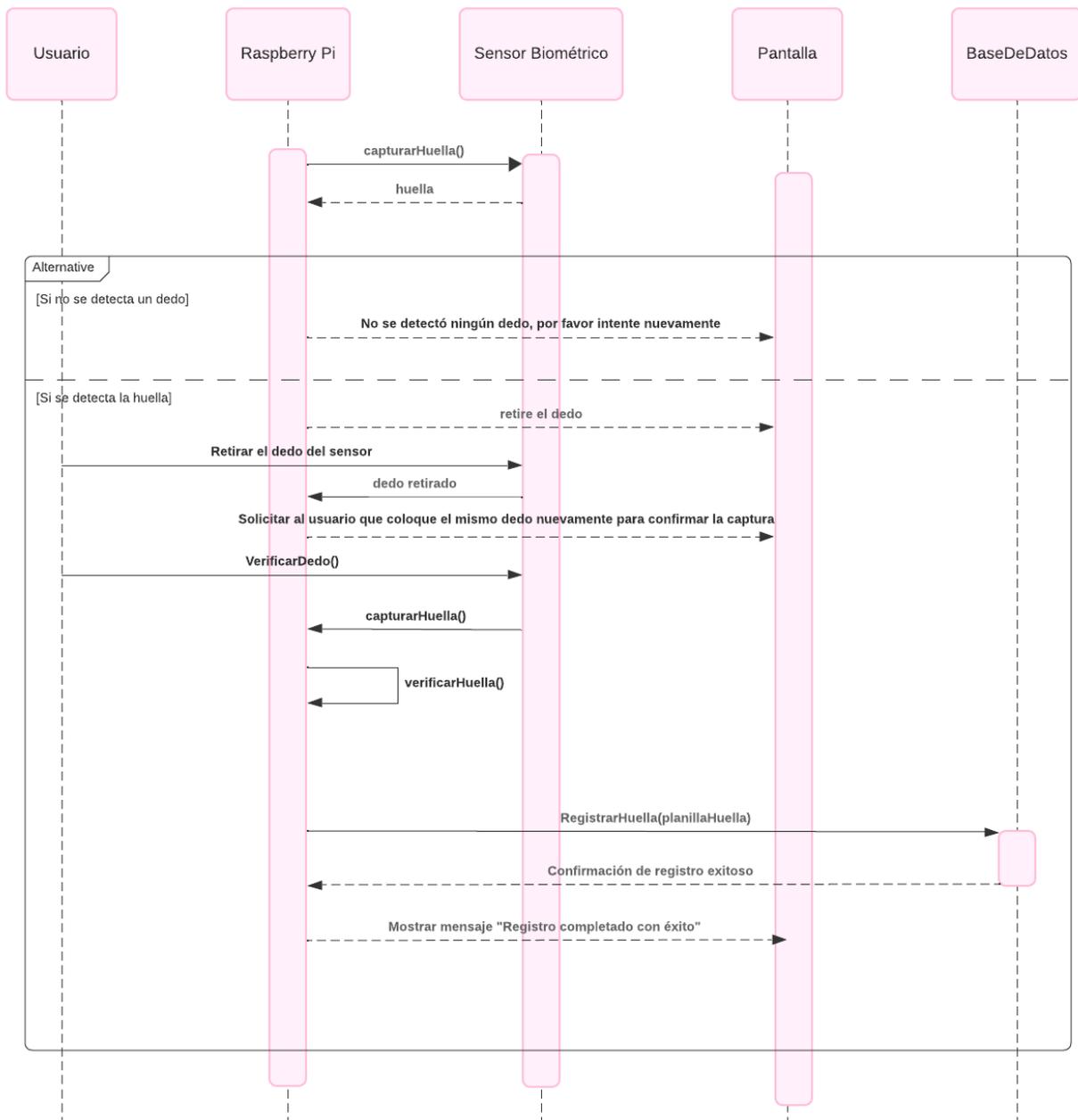


CASO DE USO (Marcar asistencia)

hay problemas de encuadre

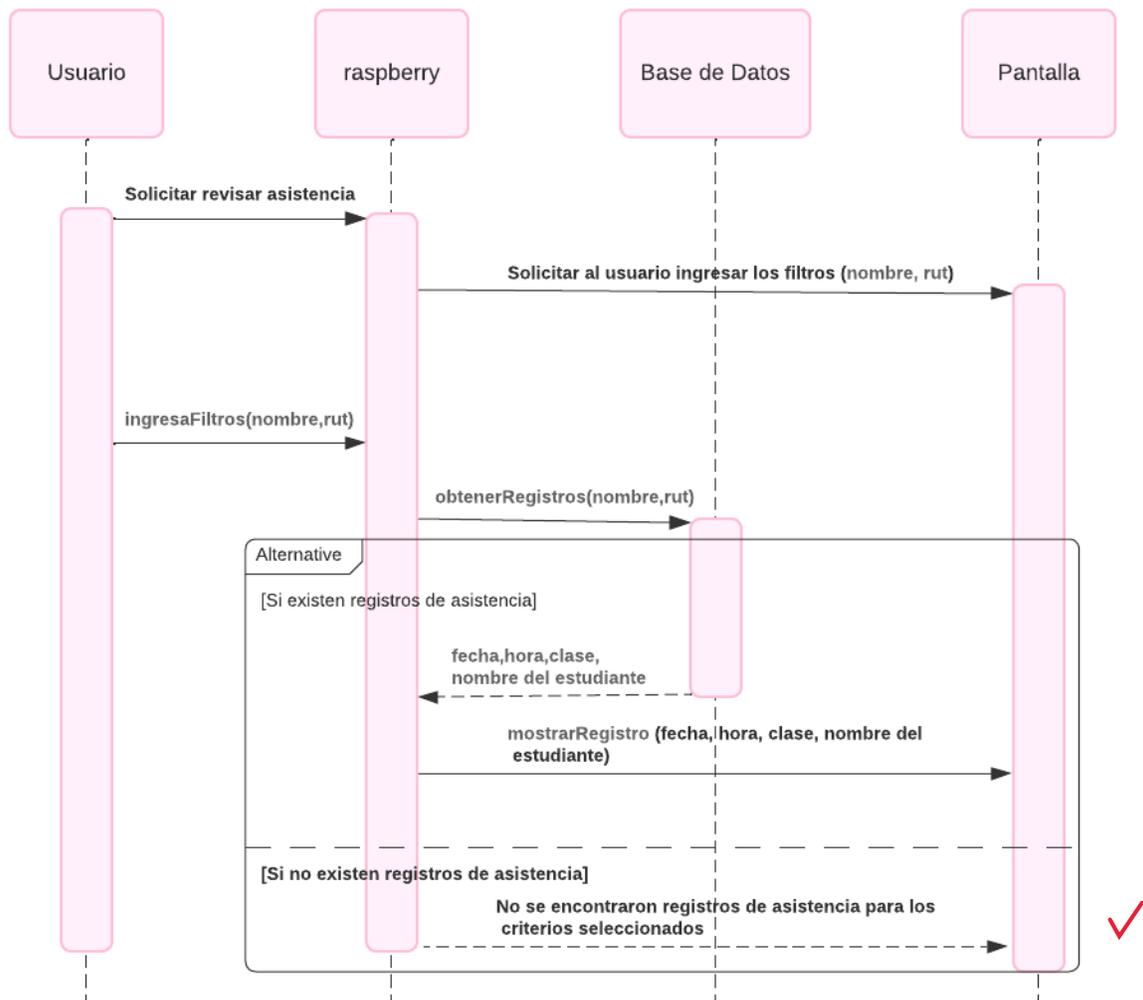
|  |   |
|--|---|
| Nombre CUS: Registrar huella dactilar.   |   |
| Descripción: Permite al estudiante registrar su huella dactilar en el sistema.   |   |
| Actor: Estudiante, sensor biométrico, raspberry pi, pantalla   |   |
| Precondición   |   |
| <p>Flujo Principal: Actor</p> <p>3. El usuario quita el dedo del sensor.</p> <p>5. El usuario ingresa el mismo dedo nuevamente en el sensor.</p> | <p>Flujo Principal: Sistema</p> <ol style="list-style-type: none"> <li>1. El sistema captura la imagen de la huella dactilar.</li> <li>2. El sistema solicita al usuario que retire el dedo.</li> <li>4. El sistema solicita al usuario que coloque el mismo dedo nuevamente para confirmar la captura.</li> <li>6. El sistema registra la huella en el sistema.</li> </ol> |
| Flujo Alternativo: "Si no se detecta un dedo"  | <ol style="list-style-type: none"> <li>1.1. El sistema muestra un mensaje en la pantalla: "No se detectó ningún dedo, por favor intente nuevamente."</li> </ol>   |
| Postcondiciones: La huella del usuario queda registrada en el sistema.   |   |
| Valor medible: Registra en el sistema los futuros usuarios de este.  |   |





|   |   |
|---|---|
| Nombre CUS: Buscar Usuario  |   |
| Descripción: permite al usuario revisar los detalles de la asistencia.  |   |
| Actor: profesor, administrador.   |   |
| Precondición: el sistema debe mantener registros válidos de asistencia previos para que el usuario pueda consultarlos.  |   |
| <p>Flujo Principal: Actor</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona la opción de ver asistencia en el sistema.</li> <br/> <li>3. El usuario ingresa los filtros.</li> </ol> | <p>Flujo Principal: Sistema</p> <ol style="list-style-type: none"> <li>2. El sistema solicita filtro para los datos(nombre,rut).</li> <br/> <li>3. El sistema muestra los registros de asistencia al usuario, incluyendo información detallada..</li> </ol> |
| Flujo Alternativo:  | <ol style="list-style-type: none"> <li>3.1. El sistema muestra un mensaje: "No se encontraron registros de asistencia para los criterios seleccionados."</li> </ol>   |
| Postcondiciones:  |   |
| Valor medible: Visualizar la información sobre las clases y la asistencia a ellas.  |   |

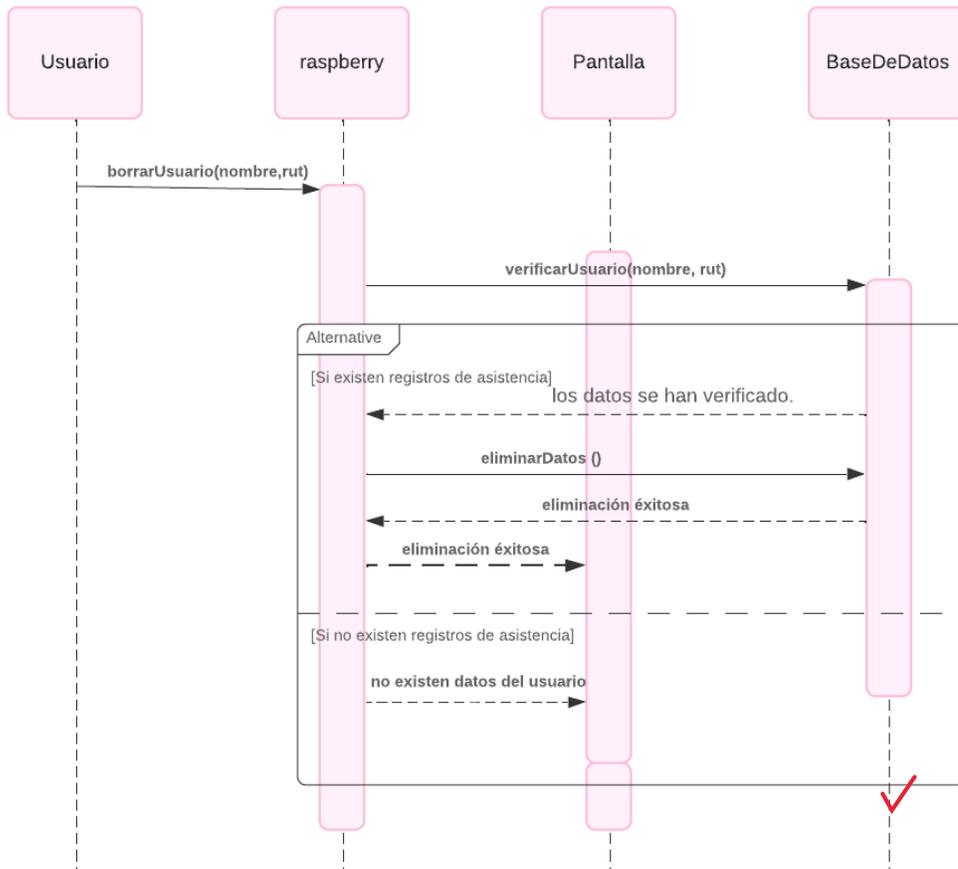




CASO DE USO (Buscar Usuario)

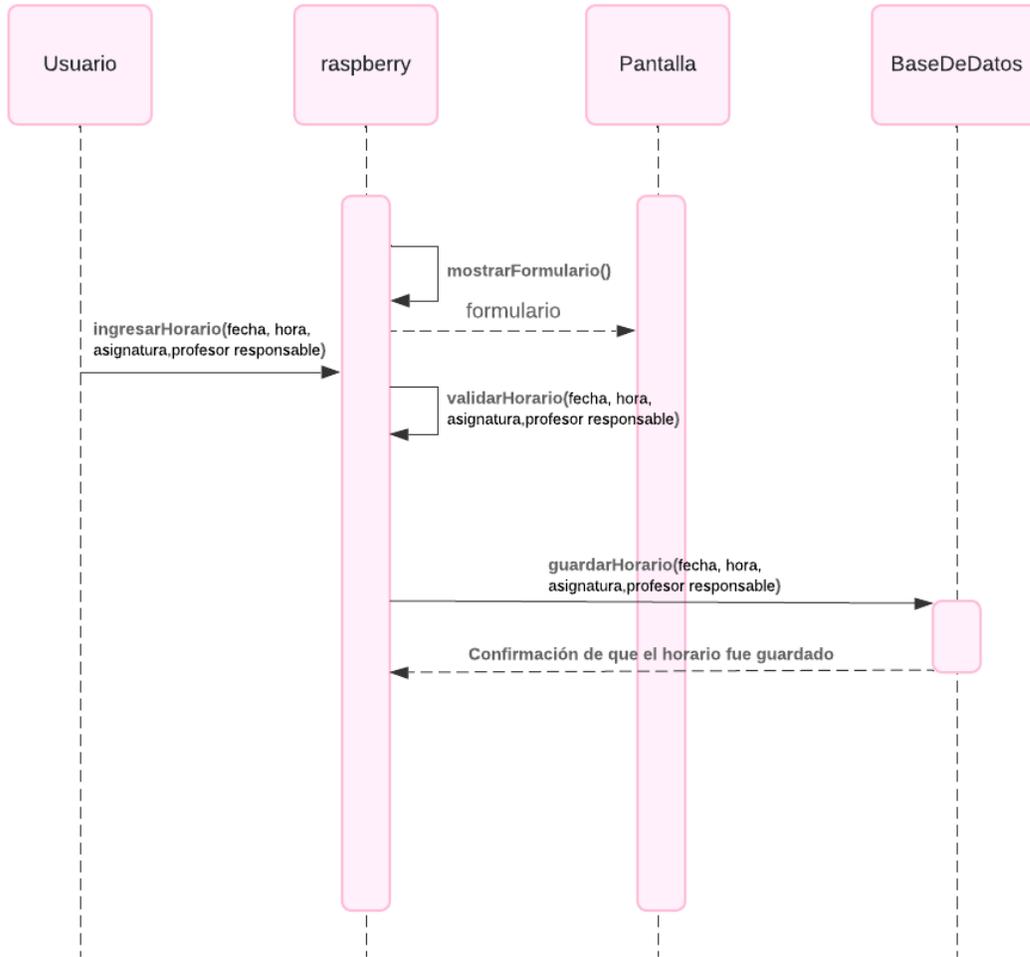
|   |   |
|---|---|
| Nombre CUS: Borrar usuario  |   |
| Descripción: Permite borrar el usuario ingresando nombre y rut  |   |
| Actor: profesor, administrador.   |   |
| Precondición: Debe existir registros de asistencia en el sistema.   |   |
| <p>Flujo Principal: Actor</p> <ol style="list-style-type: none"> <li>1. El usuario ingresa el nombre y RUT del usuario a borrar.</li> </ol> | <p>Flujo Principal: Sistema</p> <ol style="list-style-type: none"> <li>2. El sistema verifica si existen registros de asistencia asociados al usuario</li> <li>3. El sistema confirma la eliminación exitosa del usuario</li> </ol> |
| <p>Flujo Alternativo:</p> <p>No se encuentra datos del usuario</p>  | <ol style="list-style-type: none"> <li>3.1. No existen datos del usuario.</li> </ol>  |
| Postcondiciones: se actualizó la base de datos del sistema al eliminar un usuario   |   |
| Valor medible: Verificar si el usuario ha sido eliminado correctamente  |   |





|   |  |
|---|--|
| Nombre CUS: Gestionar horario   |  |
| Descripción: Permite al administrador gestionar el horario de clases, estos nos permiten saber a qué clase se toma la asistencia.   |  |
| Actor: administrador, profesor.   |  |
| Precondición:   |  |
| <p>Flujo Principal: Actor</p> <ol style="list-style-type: none"> <li>1. El actor ingresa la información del horario de clases (fecha, hora, asignatura, profesor responsable).</li> </ol> | <p>Flujo Principal: Sistema</p> <ol style="list-style-type: none"> <li>1. El sistema muestra el formulario para ingresar los datos del horario de clases</li> <li>2. El sistema valida la información ingresada (verifica formato y que no haya conflictos de horarios)</li> <li>4. El sistema guarda el horario en la base de datos.</li> </ol> |
| Flujo Alternativo:  |  |
| Postcondiciones: Queda guardado en el sistema el horario de clases.   |  |
| Valor medible: El tiempo promedio para gestionar y guardar un nuevo horario no debe superar los 5 minutos.  |  |

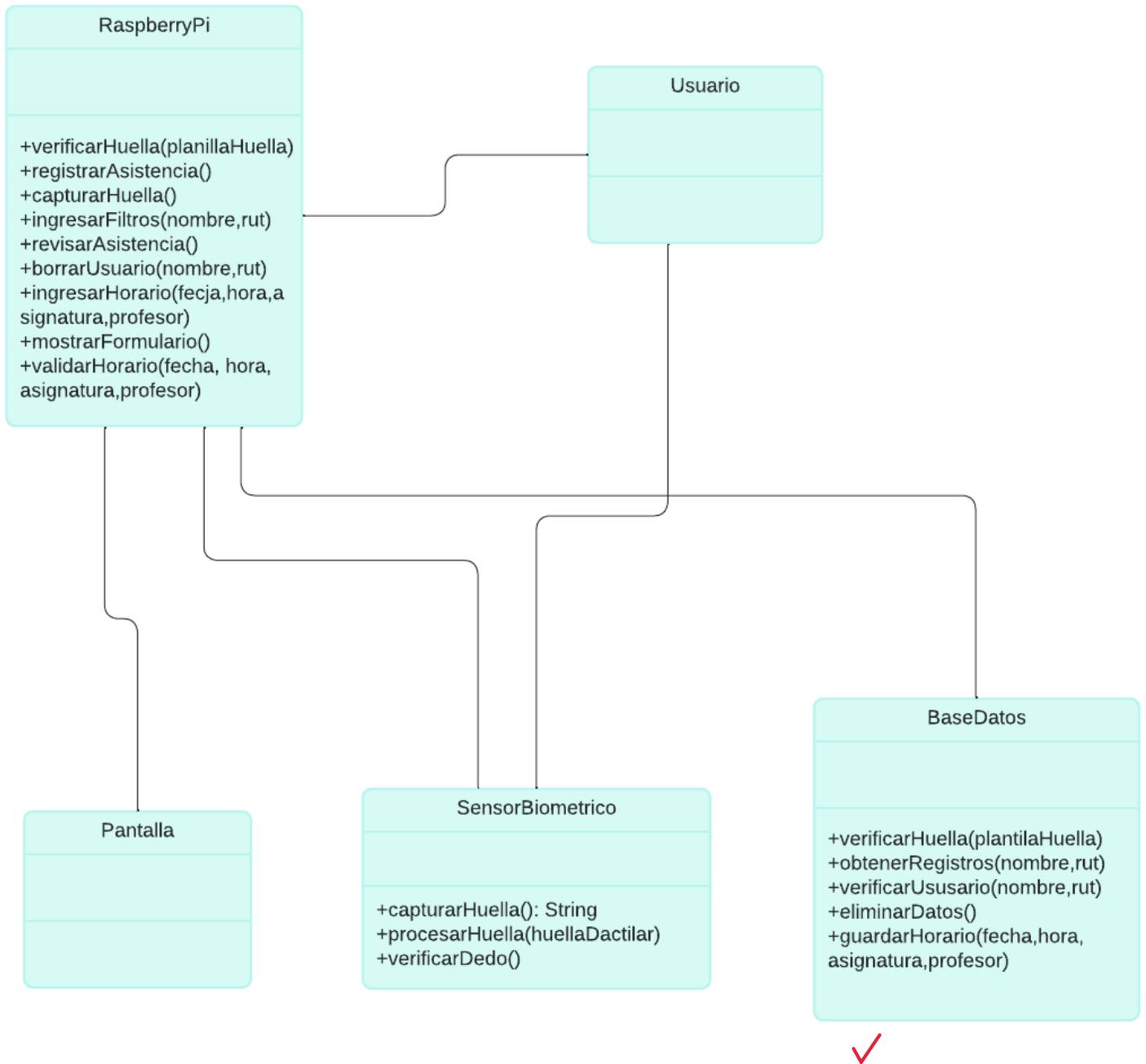




CASO DE USO (Gestionar Horario)



## Diagrama de clases



### 5.1.3. Interfaz de usuario

Logo del proyecto :



#### Bocetos de Pantallas Principales

El diseño de las pantallas del sistema de control de asistencia biométrica se desarrolló como prototipos de bajo nivel, con un enfoque minimalista para facilitar la navegación de estudiantes y profesores. Se prioriza la simplicidad y la accesibilidad, colocando los botones cerca del borde derecho para un uso cómodo.

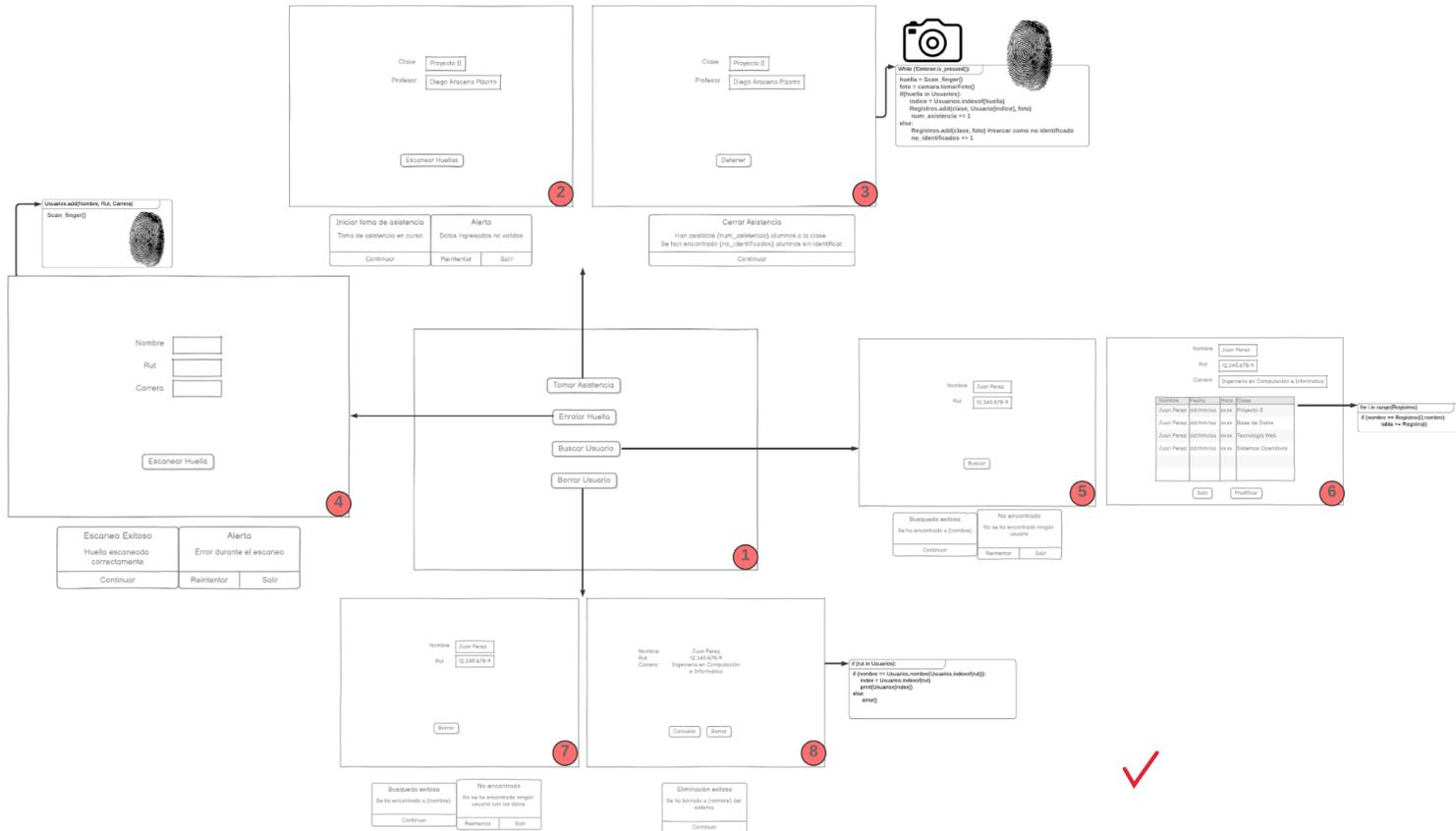
Bocetos de Pantallas:

1. Pantalla de Inicio: Acceso rápido a "Registrar Asistencia", "Revisar Asistencia" y "Configuración".
2. Pantalla de Registro de Asistencia: Área para escaneo de huella, indicador de progreso y mensajes de confirmación.
3. Pantalla de Confirmación: Mensaje de éxito con datos del registro y botón para volver al inicio.
4. Pantalla de Configuración: Opciones de personalización y acceso a soporte.

Estos prototipos aseguran una interacción intuitiva y rápida, permitiendo optimizar la experiencia de uso antes del desarrollo detallado. ✓

### Prototipo:

Se realiza un prototipo de bajo nivel del proyecto para lograr visualizar correctamente la idea a futuro para facilitar la comprensión de la estructura y flujo de ventanas para los miembros del equipo. Además de identificar problemas o mejoras en la interfaz para el momento de su realización.



1. Menú principal: con sus botones respectivos para navegar a las demás ventanas.
2. Tomar Asistencia: se le consulta al usuario el nombre de la clase y el profesor a tomar asistencia.
3. Durante esta ventana, se almacenarán todas las huellas entrantes al sensor realizando un match entre la huella recibida y las almacenadas en el sistema en conjunto a una fotografía de la persona, en caso de no identificar al usuario se marcará para una futura revisión manualmente.
4. Enrolar Huella: Se solicitará nombre, rut y carrera del usuario al momento de presionar “Escanear Huella” el sistema esperará la respuesta del sensor y escanea la huella dos veces para comprobar la correcta lectura de la huella comparando ambos intentos y procederá a almacenar la información del usuario.
5. Buscar Usuario: Se solicitará nombre y rut del usuario y al presionar el botón “Buscar” se buscará al usuario dentro del sistema para proceder con la siguiente ventana.
6. Se mostrará la información del usuario y se desplegará una tabla con todas sus ocurrencias dentro del sistema.
7. Borrar Usuario: Se solicitará nombre y rut del usuario y al presionar el botón “Borrar” se buscará al usuario dentro del sistema para proceder con la siguiente ventana.
8. Se mostrará la información del usuario y dos botones, “Cancelar” para no realizar ninguna acción y uno “Borrar” para eliminar al Usuario del sistema.

## 5.2. Herramientas y técnicas

Para poder programar en el Raspberry Pi, la lógica y tratamiento de datos utilizaremos:

- Python
- Visual Studio Code
- MySql



Para obtener los datos de los sensores utilizaremos la librería Adafruit, esta es una librería de python para poder obtener entradas de los dispositivos de entrada y salida de propósito general.

## 6. Conclusión

En esta etapa del proyecto de control de asistencia biométrica, se realizaron importantes avances en la planificación y preparación del sistema. Se definieron claramente los objetivos del proyecto y se establecieron las suposiciones y restricciones que guiarán su desarrollo. Además, se seleccionaron los componentes electrónicos esenciales, como la Raspberry Pi y el sensor Adafruit Fingerprint, garantizando que el sistema cuente con la tecnología necesaria para funcionar de manera eficiente.

También se llevaron a cabo los bocetos de las pantallas principales en prototipos de bajo nivel, permitiendo una visualización preliminar de la interfaz y ajustando el diseño para asegurar una experiencia de usuario intuitiva. Estos pasos han establecido una base sólida para continuar con el desarrollo y las pruebas del proyecto, asegurando que las siguientes fases se realicen con un enfoque bien definido y organizado.



## 7. Referencias

**Documentación de Raspberry Pi 4:** Guías y manuales oficiales para configurar y programar la [Raspberry Pi](#), esencial para el desarrollo del sistema.

**Manual de sensores Adafruit Fingerprint:** Documentación técnica que explica cómo integrar y usar el [sensor biométrico de huellas dactilares](#).

**Python Documentation:** Referencia oficial de [Python](#), el lenguaje usado en el proyecto para desarrollar el software de control de asistencia.

**Artículos académicos sobre el uso de biometría en educación:** Estudios que muestran cómo la biometría mejora la eficiencia en entornos educativos, como los publicados en el [Journal of Educational Technology Development and Exchange](#).

**GitHub Documentation:** Guías para trabajar con GitHub en la gestión del proyecto y el control de [versiones del código](#)

**Redmine Documentation:** Información sobre cómo usar [Redmine](#) para el seguimiento de tareas y gestión de proyectos, que es útil para la organización del trabajo.

**Artículos sobre mejores prácticas en desarrollo de software:** Material que ayuda a aplicar [buenas prácticas de programación](#) para garantizar un código limpio y eficiente. Ejemplo en

