**Sistema de desbloqueo de puertas con reconocimiento facial**

Especificación de Requerimientos de Software (SRS)

Santiago Catriman, Luciano Mansilla

Universidad Nacional de Río Negro

Contenido

[1. Introducción. 3](#_Toc165137663)

[1.2 Alcance 3](#_Toc165137664)

[1.3 Características de los usuarios 3](#_Toc165137665)

[1.4 Limitaciones 3](#_Toc165137666)

[2. Descripción general del sistema 4](#_Toc165137667)

[2.1 Funciones del producto 4](#_Toc165137668)

[2.2 Características de usuarios 4](#_Toc165137669)

[3. Requisitos específicos 5](#_Toc165137670)

[3.1 Requisitos funcionales 5](#_Toc165137671)

[3.2 Requisitos no funcionales 6](#_Toc165137672)

[3.3 Requisitos de software 6](#_Toc165137673)

[3.3.1 Lenguaje de programación 6](#_Toc165137674)

[3.3.2 Bibliotecas y Frameworks 6](#_Toc165137675)

[3.3.3 Plataforma de desarrollo 6](#_Toc165137676)

[3.3.4 Herramientas de desarrollo 6](#_Toc165137677)

[3.3.5 Sistema de control de versiones 6](#_Toc165137678)

[3.4 Requisitos de hardware 6](#_Toc165137679)

[3.4.1 Raspberry pi 6](#_Toc165137680)

[3.4.2 Cámara para Raspberry 7](#_Toc165137681)

[3.4.3 Cerrojo electrónico 7](#_Toc165137682)

[4. Glosario 7](#_Toc165137683)

[Bibliografía 8](#_Toc165137684)

# Introducción.

En el presente documento se detallará la Especificación de Requerimientos de Software (SRS por sus siglas en ingles) para el sistema de desbloqueo con reconocimiento facial. El proyecto sirve para aprobar la materia de sistemas embebidos de la carrera de ingeniería en computación de la Universidad Nacional de Rio Negro.

## Alcance

El sistema permitirá que una persona pueda abrir una puerta con su rostro. Incluirá, entre otras cosas, un cerrojo electrónico, una Raspberry pi 3B+ y la cámara de la Raspberry pi. Se programará con Python el código para reconocimiento facial, se realizará un proceso de entrenamiento del algoritmo con un conjunto de datos de rostros previamente cargados, si se encuentra coincidencia con el rostro que se está detectando se concederá acceso, de lo contrario no se abrirá la cerradura automática.

Para hacer las pruebas correspondientes, se decidió hacer una maqueta de una puerta de madera para mejor conveniencia.

## Características de los usuarios

En primer lugar, se encuentran los usuarios que usaran el sistema para desbloquear la cerradura, presentando su rostro a la cámara.

Por otra parte, están los administradores que serán los encargados de configurar todo el hardware y el software, de cargar los rostros en el algoritmo, etc.

## Limitaciones

El sistema no contara con una aplicación web o aplicación para teléfonos inteligentes, personas con conocimiento técnico deben cargar las imágenes de los sujetos autorizados a abrir el cerrojo.

El sistema podría fallar y no reconocer un rostro que está autorizado para ingresar.

# Descripción general del sistema

## 2.1 Funciones del producto

El sistema contara con las siguientes funciones calves:

1. Reconocimiento facial

* El sistema podrá capturar imágenes con la cámara integrada a la Raspberry pi
* Se implementará un algoritmo de reconocimiento facial que comparará las imágenes capturadas con los rostros previamente registrados en la base de datos.

1. Autenticación de usuarios

* El sistema autenticará a los usuarios comparando la imagen facial capturada con las imágenes almacenadas.
* En caso de una coincidencia exitosa, el sistema activará el cerrojo electrónico para permitir el acceso

1. Gestión de usuarios

* Los administradores del sistema podrán agregar, eliminar o actualizar usuarios autorizados para el reconocimiento facial.

## Características de usuarios

El sistema identificará y tendrá en cuenta las siguientes características de los usuarios:

1. Usuario final

* Descripción: representa a las personas que intentarán autenticarse utilizando el sistema de reconocimiento facial para obtener acceso.
* Debe ser capaz de presentarse frente a la cámara para que el sistema pueda capturar su imagen facial.
* Necesita estar autorizado y tener su rostro registrado en la base de datos del sistema para obtener acceso.

1. Administrador del Sistema

* Descripción: Este usuario tiene privilegios de administración sobre el sistema de desbloqueo con reconocimiento facial.
* Tiene la capacidad de agregar, eliminar y actualizar usuarios en la base de datos de reconocimiento facial.

Roles y Responsabilidades:

Los usuarios finales tienen la responsabilidad de seguir las instrucciones proporcionadas por el sistema durante el proceso de autenticación.

Los administradores del sistema son responsables de mantener la integridad y el funcionamiento adecuado del sistema de reconocimiento facial.

# Requisitos específicos

## Requisitos funcionales

1. Requisito Funcional RF1: Captura de imágenes faciales

* **Descripción:** El sistema debe ser capaz de capturar imágenes faciales utilizando la cámara de la Raspberry pi.
* **Entrada:** Activación por parte del usuario final para iniciar la captura de imágenes en tiempo real.
* **Salida:** Imagen facial capturada y lista para su procesamiento.
* **Actores involucrados:** Usuario final.

1. Requisito Funcional RF2:

* **Descripción:** El sistema debe procesar las imágenes capturadas para realizar el proceso de reconocimiento facial.
* **Entrada:** Imagen facial capturada.
* **Salida:** Resultado del reconocimiento (coincidencia o no con rostros previamente cargados).
* **Actores involucrados:** --

1. Requisito Funcional RF3: Autenticación de usuario

* **Descripción:** El sistema debe autenticar a los usuarios comparando la imagen facial capturada con las imágenes almacenadas.
* **Entrada:** Imagen facial capturada y base de datos con imágenes previamente cargadas
* **Salida:** Activación exitosa para que se abra el cerrojo electrónico (en caso de coincidencia exitosa)
* **Actores involucrados:** --

1. Requisito Funcional RF4: Gestión de usuarios por parte de los administradores

* **Descripción:** Los administradores deben poder agregar, eliminar o actualizar usuarios.
* **Entrada:** Imágenes de los usuarios que serán agregados o eliminados para su autorización.
* **Salida:** Confirmación de cambios realizados
* **Actores Involucrados:** Administradores

## Requisitos no funcionales

1. Requisito no funcional RNF1: La aplicación desarrollada en Python con la biblioteca OpenCV debe ser eficiente en el uso de recursos para garantizar un rendimiento óptimo en la Raspberry Pi 3B+.
2. Requisito no funcional RNF2: El código desarrollado en Python debe estar bien documentado con comentarios claros y legibles para facilitar futuras modificaciones y mantenimiento del sistema.
3. Requisito no funcional RNF3: Las pruebas en la maqueta de la puerta deben simular diferentes condiciones de iluminación y ángulos de captura para validar la precisión y robustez del sistema de reconocimiento facial.

## Requisitos de software

### 3.3.1 Lenguaje de programación

* Se utilizará el lenguaje de programación Python en su versión 3.12.0, esto se debe a su facilidad y diversas fuentes de ayuda que se pueden encontrar en la web.

### 3.3.2 Bibliotecas y Frameworks

* Se empleará la biblioteca OpenCV en la versión 4.9.0.80 para el procesamiento de imágenes y el desarrollo del algoritmo de reconocimiento facial.

### 3.3.3 Plataforma de desarrollo

* El software se implementará en una Raspberry Pi 3B+ utilizando el sistema operativo Raspbian como plataforma de ejecución.

### 3.3.4 Herramientas de desarrollo

* Utilizaremos el IDE Visual Studio Code para escribir y depurar el código de Python.

### 3.3.5 Sistema de control de versiones

* Se eligió usar Git para el CVS junto con GitHub para tener un control del código fuente.

## Requisitos de hardware

### 3.4.1 Raspberry pi

* El sistema se implementará en una Raspberry Pi 3B+
* Especificaciones:
  + Procesador: ARM Cortex A53
  + Frecuencia: 1.4 GHz
  + Números de Core: 4
  + RAM: 1GB
  + Almacenamiento: Slot para tarjeta SD
  + Conexión: Wi-Fi & Bluetooth 4.2

### Cámara para Raspberry

* Se utilizará una cámara compatible con Raspberry Pi
* Especificaciones:
* 5 MP
* Resolución: 2592x1944
* Disparo: 1/4 5M
* Grabación de 1080p a 30fps y 720p a 60fps
* Distancia Focal: 3.29

### Cerrojo electrónico

* El sistema estará integrado con un cerrojo electrónico controlado por la Raspberry Pi para permitir o denegar el acceso físico.

# Glosario

- IDE: Integrated Development Environment (Entorno de desarrollo integrado)

- CVS: Concurrent Versions System (Sistema de versiones concurrentes)

- GHz: Gigahercio

- RAM: Random Access Memory (Memoria de acceso aleatorio)

- MP: Megapixeles

- fps: Frames per second (fotogramas por segundo)

- SD: Secure Digital

# Bibliografía

LDLC. (s.f.). *LDLC*. Obtenido de Raspberry Pi 3 Model B+: https://www.ldlc.com/es-es/ficha/PB00246555.html

OpenCV. (s.f.). *OpenCV - Open Computer Vision Library*. Obtenido de OpenCV: https://opencv.org/

Python Package Index. (s.f.). *opencv-python · PyPI*. Obtenido de PyPI: https://pypi.org/project/opencv-python/

Team Asana. (14 de Enero de 2024). *Cómo redactar un documento de requisitos de software* . Obtenido de Asana: https://asana.com/es/resources/software-requirement-document-template