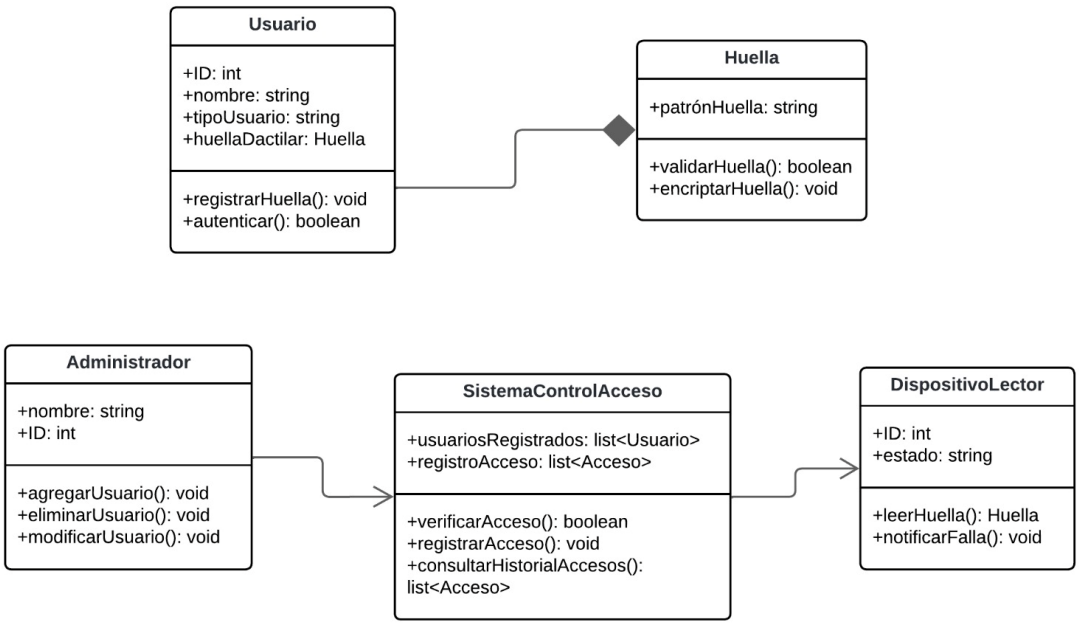
Sistema de Seguridad Basado en Huella Dactilar

Fase de Diseño

Integrantes:

- Juan Camilo Dueñas
- Keyly Pinzón
- Andrés Prada

1. Diagrama de clases



Relaciones:
Usuario - Huella: Existe una relación de composición entre Usuario y Huella, ya que cada usuario tiene una huella asociada, que es esencial para el proceso de autenticación.
Administrador - SistemaControlAcceso: El administrador gestiona el sistema, lo que le permite agregar, eliminar o modificar usuarios.
SistemaControlAcceso - DispositivoLector: El sistema se comunica con el lector para capturar las huellas y verificar su validez antes de permitir el acceso.

2. Esquema de Arquitectura del Sistema:

El sistema de seguridad basado en huella dactilar tiene una arquitectura cliente-servidor con los siguientes componentes:

Dispositivo Lector de Huellas:

- Captura la huella dactilar y la envía al sistema para su verificación.
- Se conecta con el servidor a través de una red local.

Servidor Central

- Procesa las solicitudes de autenticación comparando las huellas capturadas con la base de datos.
- Almacena de forma segura los datos biométricos en una base de datos encriptada.

Base de Datos:

- Contiene los datos de los usuarios y sus huellas en formato encriptado.
- Registra los eventos de acceso para auditorías y análisis de seguridad.

Interfaz de Usuario:

- Permite al personal administrativo gestionar usuarios (registro, eliminación, modificación).
- Permite la consulta de los registros de acceso en tiempo real.

Interacción de componentes:

1. El Dispositivo Lector captura la huella del usuario.
2. El lector envía la información al Servidor, que verifica la huella con los registros en la Base de Datos.
3. El Servidor responde al Dispositivo para permitir o denegar el acceso, y registra el evento.
4. Los administradores pueden utilizar la Interfaz de Usuario para gestionar usuarios o consultar registros.

3. Justificación del Diseño:

Decisiones de Diseño:

- **Tecnologías:** Se utilizarán lectores de huellas compatibles con los principales sistemas operativos y servidores centralizados para garantizar la escalabilidad y la gestión de miles de usuarios.

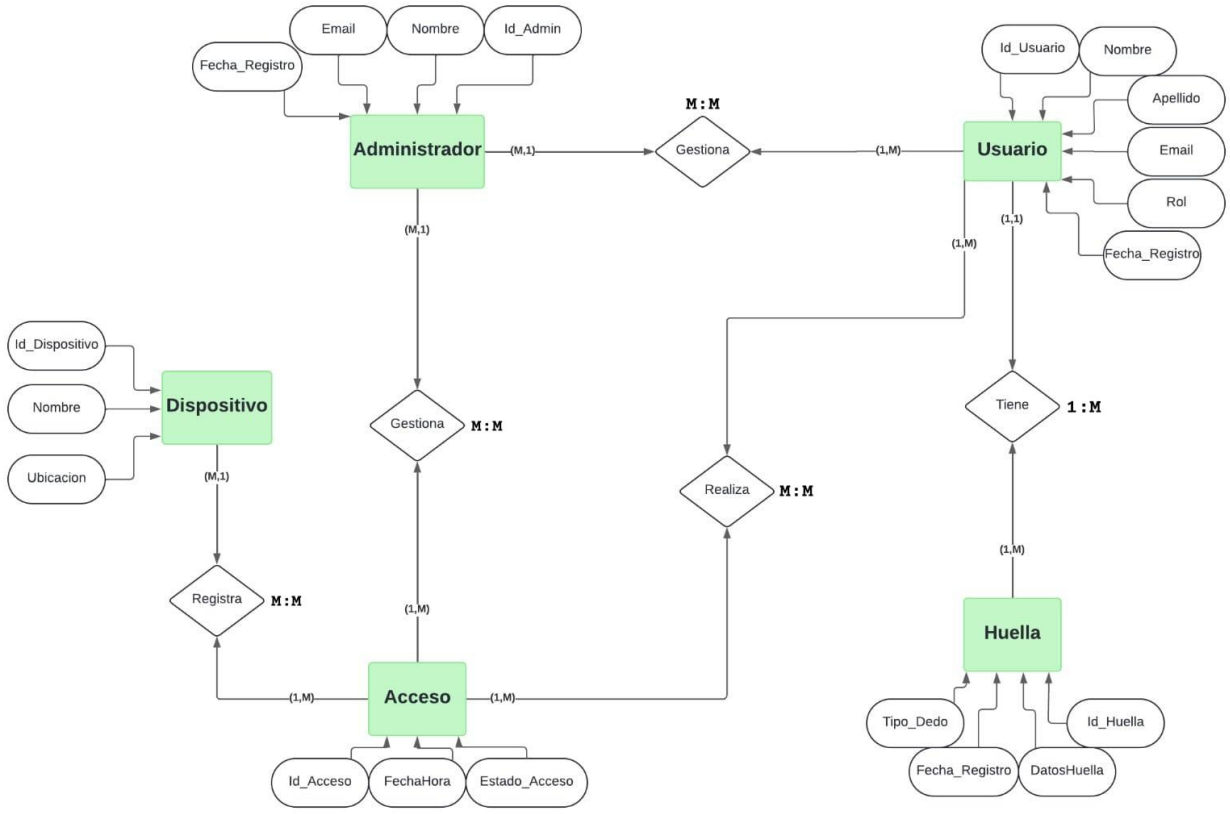
- **Base de datos**: El sistema usará una base de datos SQL debido a la necesidad de consultas rápidas y capacidad para manejar registros históricos. La información sensible se almacenará de forma encriptada.
- **Interfaces de usuario:** El sistema contará con una interfaz gráfica intuitiva para facilitar la gestión por parte del personal administrativo.

Consideraciones de Seguridad:

- **Encriptación:** Se implementará en la base de datos para asegurar la protección de los datos biométricos y evitar suplantaciones o filtraciones.

- **Monitoreo:** El sistema estará equipado con alertas ante intentos de acceso fallidos consecutivos, así como integración con cámaras de seguridad para evitar el acceso no autorizado.

2. Modelo Entidad – Relación



Modelo Entidad Relación Descrito

- Usuario es la entidad principal que interactúa con el sistema. Cada usuario tiene una Huella registrada.

- Cuando un usuario accede al sistema, se registra un Acceso. Este acceso incluye información como la hora y fecha del evento y se relaciona tanto con el Usuario como con el Dispositivo donde se realizó el acceso.

- Los Dispositivos son los que se encargan de capturar las huellas y registrarlas en el sistema.

- Administradores gestionan tanto a los usuarios como los accesos a través de una interfaz.

Diccionario de base de datos

Este diccionario de datos te permitirá desarrollar la base de datos y las consultas de manera clara y organizada, asegurando la integridad y consistencia de la información.

ENTIDAD	ATRIBUTO	TIPO DE DATO	DESCRIPCION	RESTRICCIONES
USUARIO	Id_Usuario	INT	Identificador único del usuario	PRIMARY KEY
	Nombre	VARCHAR (100)	Nombre del usuario	NOT NULL
	Apellido	VARCHAR (100)	Apellido del usuario	NOT NULL
	Email	VARCHAR (100)	Correo electrónico del usuario	UNIQUE NOT NULL
	Rol	ENUM	Rol del usuario en el sistema	NOT NULL
	Fecha_Registro	DATE	Fecha en la que el usuario fue registrado	NOT NULL

ENTIDAD	ATRIBUTO	TIPO DE DATO	DESCRIPCION	RESTRICCIONES
HUELLA	Id_Huella	INT	Identificador único de la huella dactilar	PRIMARY KEY
	DatosHuella	INT	Huella dactilar encriptada	NOT NULL
	Tipo_Dedo	ENUM	Dedo al que corresponde la huella dactilar	NOT NULL
	Fecha_Registro	DATE	Fecha en la que la huella fue registrada	NOT NULL

ENTIDAD	ATRIBUTO	TIPO DE DATO	DESCRIPCION	RESTRICCIONES
ACCESO	Id_Acceso	INT	Identificador único del registro de acceso	PRIMARY KEY
	FechaHora	DATETIME	Fecha y hora del intento de acceso	NOT NULL
	Estado_Acceso	ENUM	Resultado del intento de acceso	NOT NULL

ENTIDAD	ATRIBUTO	TIPO DE DATO	DESCRIPCION	RESTRICCIONES
DISPOSITIVO	Id_Dispositivo	INT	Identificador único del dispositivo	PRIMARY KEY
	Nombre	VARCHAR (100)	Nombre del dispositivo	NOT NULL
	Ubicación	VARCHAR (100)	Ubicación física del dispositivo	NOT NULL

ENTIDAD	ATRIBUTO	TIPO DE DATO	DESCRIPCION	RESTRICCIONES
ADMINISTRADOR	Id_Admin	INT	Identificador único del administrador	PRIMARY KEY
	Nombre	VARCHAR (100)	Nombre del administrador	NOT NULL
	Email	VARCHAR (100)	Correo electrónico del administrador	UNIQUE NOT NULL
	Fecha_Registro	DATE	Fecha en la que el administrador fue registrado	NOT NULL

3. Diseño de Interfaces

A continuación, se describe el diseño de las interfaces necesarias para la implementación del sistema.

3.1. Pantalla de Inicio de Sesión

Campos: Correo electrónico y contraseña.

Botones: "Iniciar sesión", "Recuperar contraseña".

Opciones: Rol (Usuario, Administrador).

3.2. Pantalla de Administración de Usuarios (solo para administradores)

Lista de usuarios con botones de "Agregar", "Editar" y "Eliminar".

Campos para agregar/editar usuarios: Nombre, Apellido, Email, Rol, Fecha de Registro.

3.3 Pantalla de Registro de Huellas

Selección de usuario.

Captura de huella (con notificación si la huella ya está registrada).

Fecha de registro automática.

3.4. Pantalla de Registro de Accesos

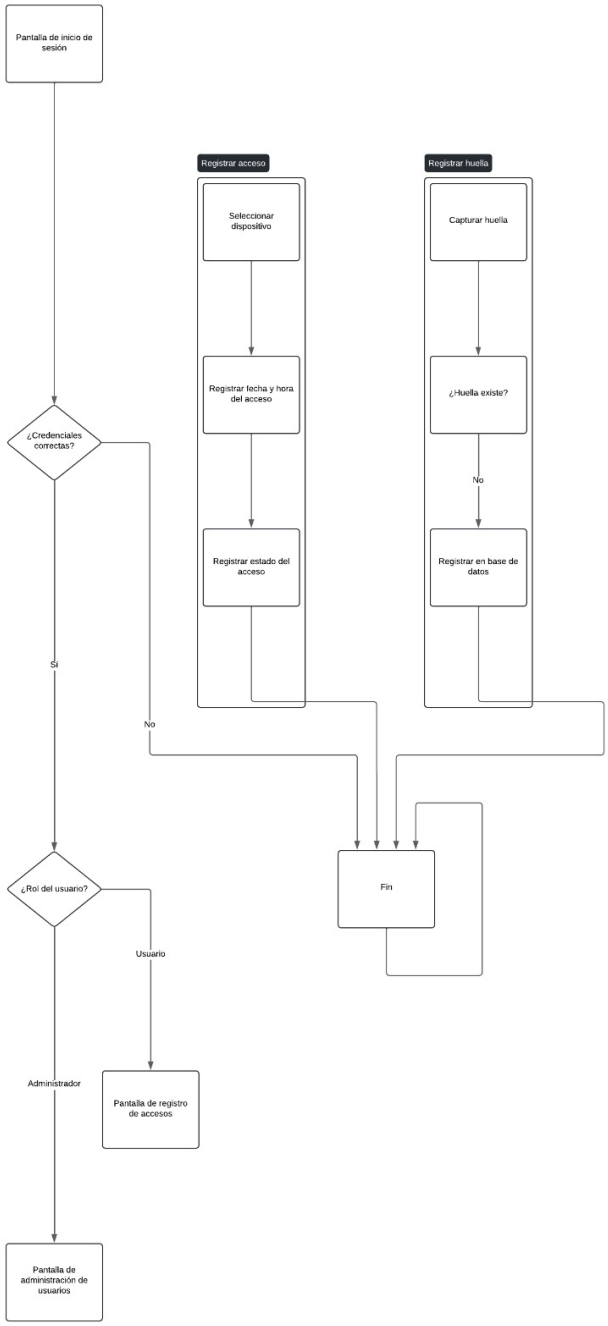
Campos: Selección de usuario, dispositivo, y fecha/hora.

Botón: "Registrar acceso".

3.5. Pantalla de Dispositivos

Lista de dispositivos registrados con opción de agregar nuevos dispositivos (Nombre, Ubicación).

4. Modelo del Algoritmo



El siguiente pseudocódigo describe el flujo principal del sistema, desde la autenticación hasta el registro de huellas y accesos.

Inicio:

Mostrar pantalla de inicio de sesión

Si credenciales correctas:

Mostrar menú principal según el rol del usuario

Si rol == 'Administrador':

Mostrar pantalla de administración de usuarios

Si rol == 'Usuario':

Mostrar pantalla de registro de accesos

Fin si

Si se selecciona 'Registrar huella':

Capturar huella

Si la huella no existe:

Registrar en la base de datos con el ID del usuario

Fin si

Fin si

Si se selecciona 'Registrar acceso':

Seleccionar dispositivo

Registrar fecha y hora del acceso

Registrar estado del acceso (éxito o fallo)

Fin si

Fin

5. Arquitectura de Red

La arquitectura de red debe estar diseñada para garantizar una comunicación eficiente y segura entre los distintos componentes del sistema.

5.1 Componentes de la Red:

Dispositivos de lectura de huella dactilar: Colocados en las entradas y salidas de la universidad, conectados a la red a través de cableado Ethernet o de manera inalámbrica (Wi-Fi, si es seguro y estable).

Servidores de base de datos: Un servidor dedicado que almacena de manera centralizada la información biométrica (huellas dactilares) y los registros de acceso.

Servidor de aplicaciones: Donde se ejecuta el software que gestiona el sistema de control de acceso, incluyendo la interfaz de administración y el procesamiento de los datos de autenticación.

Cámaras de vigilancia: Integradas al sistema de monitoreo, activadas en caso de accesos denegados o intentos sospechosos.

Estaciones de administración: Computadoras o dispositivos utilizados por el personal administrativo para gestionar el sistema, como registrar o dar de baja usuarios.

5.2 Topología de la Red:

Topología estrella: Se recomienda utilizar una topología de red estrella, donde todos los dispositivos de lectura de huella, cámaras y estaciones de administración están conectados a un switch central o a un router, que a su vez se conecta al servidor de aplicaciones y bases de datos. Esta configuración permite centralizar el control y facilita la expansión futura del sistema.

5.3 Comunicación y protocolos:

Conexión cableada (Ethernet): Para garantizar un funcionamiento estable y reducir riesgos de interferencias, las estaciones de autenticación (lectores de huella) deben conectarse por cable Ethernet. La velocidad de transmisión debe ser suficiente para garantizar una respuesta rápida (1 Gbps es recomendable).

Conexión inalámbrica (Wi-Fi): Si se decide utilizar Wi-Fi en ciertos puntos, la red debe configurarse con estándares seguros como WPA3.

VPN (Virtual Private Network): Para permitir acceso remoto seguro a la administración del sistema desde fuera de la universidad, se puede implementar una VPN que encripte el tráfico.

5. Arquitectura de Seguridad:

La arquitectura de seguridad debe ser robusta para proteger tanto los datos sensibles (huellas dactilares) como la infraestructura de red del sistema.

5.1 Seguridad de la Base de Datos

Encriptación de datos: La información biométrica (huellas) debe ser encriptada antes de ser almacenada en la base de datos. Se recomienda usar encriptación asimétrica (RSA o ECC) o simétrica (AES de 256 bits) para proteger los datos.

Autenticación de acceso: Solo personal autorizado podrá acceder a la base de datos. Se debe implementar un sistema de autenticación fuerte, como autenticación multifactor (MFA), para proteger el acceso al servidor.

5.2 Seguridad en la Transmisión de Datos

Cifrado de la comunicación: Toda la información que se transmita entre los dispositivos (lectores de huella) y el servidor de aplicaciones debe estar cifrada mediante el protocolo TLS (Transport Layer Security), garantizando que no pueda ser interceptada o manipulada.

Certificados digitales: Los servidores y los dispositivos deben utilizar certificados digitales para autenticar y cifrar las conexiones de red.

5.3 Control de Acceso

Roles y permisos: La plataforma debe implementar control de acceso basado en roles (RBAC), donde se asignen permisos específicos según el rol (administrador, personal de seguridad, estudiante).

Registro de actividad (logs): El sistema debe registrar todas las acciones realizadas por los usuarios con permisos administrativos y auditar intentos fallidos de acceso para detectar cualquier anomalía.

5.4 Protección contra amenazas externas

Firewall: Es fundamental implementar un firewall tanto a nivel del servidor como en el perímetro de la red de la universidad, para bloquear intentos de acceso no autorizados y proteger contra ataques externos como DDoS.

Sistema de detección de intrusos (IDS/IPS): Un sistema IDS/IPS se encargará de monitorear el tráfico en busca de patrones sospechosos o intentos de intrusión, alertando al equipo de seguridad para que actúe de manera inmediata.

5.5 Backup y recuperación de datos

Respaldos periódicos: La base de datos debe tener un sistema de backup automatizado y redundante, almacenando copias de seguridad en una ubicación separada (incluso en la nube) para garantizar la recuperación en caso de fallos o incidentes.

Plan de recuperación ante desastres (DRP): Debe establecerse un plan que detalle cómo actuar en caso de fallo total del sistema o pérdida de datos, garantizando una rápida recuperación y minimizando la interrupción del servicio.

6. Sistemas Gestores

Para el proyecto de control de acceso basado en huella dactilar en la universidad, es fundamental seleccionar los sistemas gestores adecuados para garantizar el buen funcionamiento, la seguridad y la escalabilidad del sistema.

6.1 Sistemas Gestores de Bases de Datos (SGBD)

El SGBD será crucial para almacenar de manera eficiente y segura la información relacionada con los usuarios, incluyendo las huellas dactilares y los registros de acceso.

PostgreSQL: Es una opción muy robusta y escalable, además de ser de código abierto. Ofrece buen rendimiento y seguridad, y tiene soporte para manejar grandes volúmenes de datos y transacciones. También permite el uso de funciones de encriptación avanzadas para proteger los datos biométricos.

6.2 Sistema Gestor de Identidades y Accesos (IAM)

Un Sistema Gestor de Identidades y Accesos es clave para controlar quién tiene acceso a qué partes del sistema, tanto a nivel de usuarios finales (estudiantes, personal) como del equipo administrativo que gestiona la plataforma.

Opciones:

Active Directory (AD): Si la universidad ya utiliza un entorno de Microsoft, integrar Active Directory (junto con Azure AD si es necesario) puede ser una opción para gestionar la identidad y permisos de los usuarios. Permite controlar roles y autenticar usuarios de forma segura.

OpenLDAP: Si buscas una solución de código abierto, OpenLDAP es una opción para gestionar las identidades de los usuarios de forma centralizada. Soporta múltiples plataformas y es altamente configurable.

6.3 Sistema Gestor de Seguridad de Información (SGSI)

El Sistema Gestor de Seguridad de la Información asegura que las políticas de seguridad sean implementadas y mantenidas adecuadamente, protegiendo la información sensible (como huellas dactilares) y garantizando el cumplimiento de normativas legales.

Opciones:

ISO 27001 Framework: Implementar este estándar es ideal para establecer un SGSI que cubra todos los aspectos de la seguridad de la información en el proyecto, desde el acceso hasta la protección de los datos biométricos. Aunque ISO 27001 no es un sistema en sí, es un marco normativo que te permitirá definir y gestionar controles de seguridad.

SIEM (Security Information and Event Management): Integrar un sistema de gestión de eventos de seguridad (SIEM) para monitorear los registros de acceso, las alertas y

posibles amenazas en tiempo real. Ejemplos de SIEM son Splunk o ELK Stack (Elasticsearch, Logstash, Kibana).

6.4 Sistema Gestor de Dispositivos (MDM)

Dado que estarás utilizando múltiples dispositivos de lectura de huella en las entradas de la universidad, será importante contar con un Sistema Gestor de Dispositivos para mantener el control sobre estos dispositivos, asegurar su correcto funcionamiento y gestionarlos de manera remota.

Opciones:

Microsoft Intune: Si ya estás utilizando infraestructura de Microsoft, Intune te permite gestionar los dispositivos de manera centralizada, asegurando que estén configurados correctamente y actualizados.

MobileIron: Es una opción flexible para la gestión de dispositivos IoT y sensores biométricos que puede adaptarse a entornos más diversos.

6.5 Sistema Gestor de Mantenimiento y Soporte

Es recomendable implementar un Sistema de Gestión de Mantenimiento que te permita monitorear y gestionar el estado de los dispositivos de lectura de huella, las cámaras de seguridad y el software en general. Esto asegura que cualquier problema técnico sea detectado y resuelto rápidamente.

Opciones:

ServiceNow: Ofrece un sistema de gestión de incidentes que puede ser usado para manejar problemas técnicos o realizar tareas de mantenimiento preventivo en el sistema.

OTRS: Una solución de código abierto que puede servir para gestionar solicitudes de soporte, incidentes o fallos en el sistema de control de acceso.

6.6 Sistema Gestor de Registros (Logs)

El sistema debe registrar todos los accesos, intentos fallidos y cualquier cambio administrativo. Implementar un Sistema Gestor de Logs garantizará que estos datos se almacenen de manera segura y se puedan auditar en caso necesario.

Opciones:

Graylog: Es una solución de código abierto que te permite centralizar, analizar y auditar todos los registros generados por el sistema de acceso.

Splunk: Ofrece una plataforma robusta para la gestión y análisis de logs en tiempo real, facilitando la identificación de eventos sospechosos o fallos en el sistema.

6.7 Sistema Gestor de Actualizaciones (Patch Management)

Dado que el sistema involucrará múltiples componentes de software y hardware, es fundamental implementar un Sistema Gestor de Actualizaciones para mantener todos los dispositivos y servidores actualizados con los últimos parches de seguridad.

Opciones:

WSUS (Windows Server Update Services): Si el sistema se implementa en un entorno Windows, WSUS permite gestionar actualizaciones de manera centralizada.

Chef/Puppet/Ansible: Para sistemas más variados o basados en Linux, estas herramientas de automatización permiten gestionar actualizaciones y configuraciones a gran escala.

7. Lenguajes de Programación:

7.1 Backend:

Python (Django o Flask): Python es excelente para el desarrollo rápido y la integración con bibliotecas de procesamiento biométrico. Django proporciona un framework robusto con muchas funcionalidades integradas (autenticación, manejo de bases de datos), mientras que Flask es más ligero y adecuado para construir APIs RESTful.

Bibliotecas: Usaremos PyFingerprint o Biopython para la parte biométrica, ya que permiten el procesamiento y la verificación de huellas dactilares.

7.2 Frontend:

HTML5, CSS3, JavaScript (con React.js o Vue.js): HTML5/CSS3 es esencial para la creación de la interfaz de usuario. React.js o Vue.js se usarán para crear una interfaz de administración dinámica y eficiente. Ambas opciones ofrecen una gran flexibilidad para construir interfaces que gestionen los usuarios y muestren los registros de accesos de manera clara y en tiempo real.

7.3 Procesamiento Biométrico:

Python o C++: Python es ideal para integrar a los lectores de huellas con el sistema. Su sintaxis sencilla y las bibliotecas específicas hacen que sea fácil trabajar con dispositivos biométricos. C++ podría usarse si se requiere un procesamiento más cercano al hardware o para optimizar el rendimiento de la captura de huellas.

8. Herramientas de Apoyo

8.1. Herramientas de Gestión de Proyectos

Microsoft Project: Ideal para proyectos más grandes y estructurados. Permite planificar cronogramas detallados, asignar recursos y hacer un seguimiento del avance del proyecto.

8.2 Herramientas de Modelado de Arquitectura y Diagramas

Para representar visualmente la arquitectura de red, la seguridad, el flujo de datos y los componentes del sistema, estas herramientas son clave.

Lucidchart: Una herramienta muy popular para crear diagramas de flujo, arquitecturas de red, mapas conceptuales, diagramas UML y mucho más. Es intuitiva y permite la colaboración en tiempo real.

Microsoft Visio: Una solución más completa y corporativa, Visio es excelente para diagramar arquitecturas, flujos de trabajo, redes y más. Es parte del ecosistema de Microsoft Office, por lo que tiene buena integración si ya se usan otras herramientas de Microsoft.

ERDPlus: Para la creación de modelos de bases de datos (diagrama entidad-relación), especialmente útil al definir la estructura de la base de datos del proyecto.

8.3 Herramientas de Diseño y Prototipado

Útiles para realizar un prototipo de la interfaz de usuario o hacer una presentación visual de cómo funcionaría el sistema.

Figma: Una herramienta basada en la nube para el diseño de interfaces de usuario y prototipos interactivos. Permite diseñar cómo se vería el sistema de administración y el acceso de usuarios.

Adobe XD: Similar a Figma, pero con más integración al ecosistema de Adobe. Es ideal para crear prototipos interactivos y hacer pruebas de usabilidad antes de desarrollar el software.

8.4. Herramientas de Pruebas y Simulación

Para verificar que el sistema de control de acceso funciona correctamente y cumple con los requisitos, las herramientas de simulación y pruebas permiten validar el diseño y corregir posibles fallos.

Wireshark: Útil para monitorear y analizar el tráfico de red. Ayudará a asegurar que las comunicaciones entre los dispositivos de lectura de huellas y el servidor estén funcionando correctamente y sean seguras.

Apache JMeter: Ideal para realizar pruebas de rendimiento en el sistema, especialmente para simular cómo funcionará con muchos usuarios. Permite evaluar la capacidad de respuesta del sistema y su estabilidad bajo carga.

8.5. Herramientas de Control de Versiones y Colaboración

Es importante que todo el equipo tenga acceso al código y pueda colaborar en el desarrollo del proyecto de manera organizada y controlada.

GitHub: La plataforma más popular para control de versiones. Permitirá gestionar el código fuente, realizar revisiones, colaborar con el equipo y gestionar el desarrollo de software de manera eficiente.

8.6. Herramientas de Seguridad y Auditoría

Para garantizar la seguridad del sistema desde el desarrollo hasta la implementación, estas herramientas serán esenciales para detectar y mitigar vulnerabilidades.

OWASP ZAP: Una herramienta gratuita de código abierto para realizar pruebas de penetración en aplicaciones web. Ayuda a identificar vulnerabilidades en la aplicación del sistema de control de acceso.

Burp Suite: Similar a OWASP ZAP, pero más avanzado y con funcionalidades adicionales para realizar pruebas de seguridad más profundas.

Nessus: Ideal para realizar escaneos de vulnerabilidades en la red y los dispositivos conectados, como los lectores de huellas dactilares.

8.7. Herramientas de Automatización de Implementación (CI/CD)

Si deseas automatizar el proceso de desarrollo, pruebas e implementación del sistema, estas herramientas te permiten crear un flujo de trabajo ágil y eficiente:

Jenkins: Una herramienta de integración continua (CI) y entrega continua (CD) que te permitirá automatizar las pruebas y despliegue del sistema.

CircleCI: Similar a Jenkins, pero con mayor facilidad de configuración y totalmente en la nube. Útil para pequeños equipos que no desean preocuparse por la infraestructura.

Docker: Herramienta de contenedorización que permite aislar las aplicaciones y sus dependencias para hacer más fácil su despliegue en diferentes entornos.

8.8 Herramientas de Documentación

La documentación es clave en cualquier proyecto. Estas herramientas ayudarán a crear y mantener la documentación del proyecto de manera organizada.

Confluence: Excelente para documentar todo el proceso del proyecto, desde la planificación hasta la implementación. Está diseñado para equipos y ofrece funcionalidades colaborativas.

Notion: Una herramienta todo-en-uno que permite organizar notas, tareas, bases de datos y más. Es una opción flexible y sencilla para documentar cada fase del proyecto.

Microsoft Word/OneNote: Si se prefiere usar herramientas del ecosistema de Microsoft, tanto Word como OneNote son opciones para documentar el proyecto de manera formal y compartirla con otros colaboradores.