Escuela Superior
de Ingeniería y Tecnología
Universidad de La Laguna

Trabajo de Fin de Grado

Estudio de sistema de control de accesos basado en códigos ópticos y propuesta de solución basada en aplicación web

Study of an Access Control System Based on Optical Codes and Proposal of a Solution Based on a Web Application

Laura Dorta Marrero

La Laguna, 17 de junio de 2024

D. **Cándido Caballero Gil**, profesor Titular de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor.

D. **Julio Francisco Rufo Torres**, Investigador Ramón y Cajal de Universidad adscrito al Departamento de Ingenieria Industrial, como cotutor.

C E R T I F I C A (N)

Que la presente memoria titulada:

”*Estudio de sistema de control de accesos basado en códigos ópticos y propuesta de solución basada en aplicación web*”

ha sido realizada bajo su dirección por **Laura Dorta Marrero**.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 17 de junio de 2024

Agradecimientos

Quiero expresar mi profundo agradecimiento a todas las personas que han hecho posible la culminación de este Trabajo de Fin de Grado. Sus contribuciones, apoyo y aliento han sido fundamentales a lo largo de este proceso académico.

Primero y ante todo, quiero agradecer a mi tutor, Cándido Caballero Gil, por su orientación y paciencia. Sus consejos han ayudado a dar forma a este trabajo.

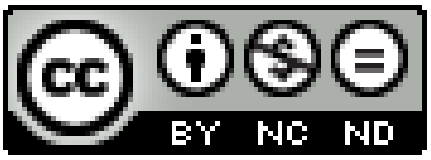
Además, quiero agradecer sinceramente a mi cotutor, Julio Francisco Rufo Torres, por su asesoramiento y por brindarme perspectivas clave que han enriquecido mi trabajo.

Agradezco también a mis compañeros de clase y amigos, quienes han ofrecido su apoyo moral y compartido valiosas ideas a lo largo de este viaje académico.

No puedo olvidar mencionar a mi familia por su amor incondicional, comprensión y constante estímulo. Su apoyo ha sido el pilar sobre el cual he construido este proyecto.

A todos ellos, mi más sincero agradecimiento.

Licencia



Resumen

Los sistemas de control de acceso físico (PAC) son herramientas cruciales para gestionar y restringir la entrada a espacios físicos mediante el uso de credenciales, lectores, puntos de acceso y paneles de control. Estos sistemas varían en niveles de seguridad, funcionalidades y costes. El objetivo principal de este trabajo es mejorar la seguridad de uno de estos sistemas. En particular, se enfocará en optimizar la seguridad de una cerradura desarrollada por el cotutor, la cual utiliza tecnología de luz visible (VLC) que se basa en la utilización de pulsos de luz para transmitir datos.

La cerradura emplea receptores ópticos para detectar la secuencia de cambios ópticos que conforman la llave de acceso. Esta secuencia, transmitida a través de pulsos de luz, añade una capa adicional de complejidad y seguridad al sistema, dificultando las posibilidades de acceso no autorizado. El trabajo abordará la creación de una aplicación web que facilite la comunicación de la clave a los usuarios.

Palabras clave: TFG, Acceso Físico, Seguridad,Comunicación de Luz Visible, VLC, Control de acceso.

Abstract

Physical Access Control (PAC) systems are crucial tools for managing and restricting entry to physical spaces using credentials, readers, access points, and control panels. These systems vary in levels of security, functionality, and cost. The main objective of this project is to improve the security of a particular system. The focus will be on enhancing the security of a lock created by the co-supervisor, which utilizes visible light communication (VLC) technology. This technology involves transmitting data using light pulses.

The lock employs optical receivers to detect the sequence of optical changes constituting the access key. This sequence, transmitted through light pulses, adds an additional layer of complexity and security to the system, making unauthorized access more difficult. The work will address the creation of a web application to facilitate the communication of the key to users.

Keywords: TFG, Physical Access, Security, Visible Light Communication, VLC, Access Control.

Índice general

Índice de Figuras

Capítulo 1

Introducción

En el ámbito de las viviendas y otros espacios equipados con cerraduras electrónicas, la seguridad y la accesibilidad son aspectos cruciales que afectan directamente la experiencia de los usuarios. La complejidad en el manejo de claves para acceder a estos lugares puede generar situaciones de estrés. Con el objetivo de mejorar esta situación, este proyecto se enfoca en el desarrollo de una aplicación diseñada para simplificar el proceso de gestión de accesos, asegurando una experiencia intuitiva y segura que optimice tanto la seguridad como la comodidad de los usuarios en diversos contextos, como alojamientos turísticos, residencias personalizadas y empresas.

1.1. Contexto y Justificación

Por diversas razones, estas cerraduras resultan un desafío para los usuarios, especialmente cuando son múltiples personas las que necesitan acceder. A menudo, no saben cómo compartir la clave de manera segura o cómo utilizar las aplicaciones existentes. Este problema es particularmente grave para ciertos grupos demográficos como las personas de edad avanzada.

La complejidad de las cerraduras modernas puede generar situaciones de inseguridad y estrés. Por ejemplo, los usuarios mayores pueden tener dificultades para manejar aplicaciones en dispositivos inteligentes. Esto puede resultar en que se queden fuera de sus propias viviendas, lo que no solo es incómodo, sino potencialmente peligroso si ocurre en situaciones de emergencia o durante horas nocturnas. Además, el uso incorrecto o la mala configuración de las aplicaciones de cerraduras puede llevar a vulnerabilidades de seguridad, como la posibilidad de que personas no autorizadas obtengan acceso.

En el contexto de las viviendas vacacionales, estas complicaciones pueden ser aún más pronunciadas. Los turistas que alquilan estas propiedades pueden enfrentarse a sistemas de cerraduras desconocidos y no intuitivos, aumentando su riesgo de quedarse fuera en un lugar que no conocen o de comprometer la seguridad de la propiedad. La falta de una manera fácil y segura de compartir la clave entre los miembros de un grupo de viajeros puede causar frustraciones y problemas logísticos.

Por otro lado, en viviendas personalizadas donde se emplean estas tecnologías, los residentes y sus visitantes también pueden encontrar dificultades similares. La inseguridad y la ansiedad pueden incrementarse si no se puede acceder rápidamente al hogar, especialmente en situaciones de emergencia médica o en momentos críticos.

Por todo esto, este proyecto se centra en simplificar el proceso de transmisión de claves. Una solución intuitiva y fácil de usar es necesaria para mejorar la experiencia de los usuarios y garantizar su seguridad y tranquilidad. Aunque la necesidad es actualmente más urgente en viviendas vacacionales y personalizadas, esta solución también podría facilitar la adopción de cerraduras modernas en hoteles y empresas, proporcionando un método de acceso seguro y eficiente.

1.2. Objetivos

Mejora de la Experiencia del Usuario: La aplicación se diseñará con una interfaz intuitiva y amigable, garantizando que los usuarios puedan gestionar accesos de forma fácil y eficiente. Se priorizará la usabilidad para asegurar una experiencia fluida desde la instalación hasta el uso diario.

Incremento de la Seguridad: Se implementará un sistema avanzado de generación y gestión de claves de cerraduras VLC, utilizando métodos de cifrado seguros y un protocolo seguro de envío por correo electrónico. Esto reducirá significativamente el riesgo asociado con la pérdida o el robo de llaves físicas.

Optimización en la Gestión de Accesos: La aplicación permitirá una gestión centralizada y eficaz de accesos, facilitando la creación y revocación de claves temporales, así como la supervisión remota de accesos desde dispositivos móviles. Esto optimizará la administración operativa y mejorará la capacidad de respuesta ante cambios en la

configuración de accesos.

1.3. Metodología

En este proyecto, no se ha seguido una metodología formalmente estructurada. En cambio, se optó por un enfoque basado en la colaboración continua y la comunicación frecuente entre los interesados. Este enfoque incluyó reuniones prácticamente semanales, durante las cuales se revisaron los avances, se resolvieron dudas y se planificaron los siguientes pasos.

El proyecto se dividió en cuatro fases importantes:

1. **Estudio de tecnologías:** En esta fase inicial del proyecto, se llevó a cabo una exhaustiva investigación y evaluación de diversas tecnologías tanto para el desarrollo de software como para la integración con el hardware específico, como la cerradura VLC sobre la cual se desarrollaría la aplicación. En términos de software, esto implicó explorar frameworks, herramientas de desarrollo, plataformas de hosting y otras tecnologías relevantes. El objetivo primordial fue seleccionar las soluciones tecnológicas más adecuadas que cumplieran con los requisitos y objetivos del proyecto. Esta fase resultó crucial para establecer las bases tecnológicas sobre las cuales se construiría la aplicación.

2. **Adaptación de tecnologías al software:** Una vez seleccionadas las tecnologías adecuadas, se procedió a adaptarlas y configurarlas según las necesidades específicas del proyecto. Esto implicó el desarrollo de la interfaz web, que actúa como el medio principal para transmitir la clave a la cerradura. Durante esta etapa, se diseñaron y desarrollaron los componentes visuales y funcionales necesarios para interactuar de manera efectiva con el sistema de cerraduras.

3. **Desarrollo de la aplicación funcional:** En esta fase, el equipo se centró en el desarrollo del backend y el frontend de la aplicación. El backend se encarga de la gestión de datos y la comunicación con la base de datos y otros sistemas. Mientras tanto, el frontend se ocupa de la interfaz de usuario, asegurando que sea fácil de usar y funcional.

4. **Documentación:** Esta última fase implica la elaboración de la memoria, que consiste en un informe exhaustivo detallando el desarrollo

completo del proyecto hasta su conclusión. Este documento incluye la descripción de los objetivos, metodologías empleadas, resultados obtenidos, conclusiones alcanzadas y posibles recomendaciones para proyectos similares en el futuro.

1.3.1. Ventajas del Enfoque Adoptado

- Adaptabilidad: La flexibilidad del enfoque ha permitido ajustarse rápidamente a cambios y desafíos emergentes.
- Comunicación efectiva: Las reuniones frecuentes facilitaron una comunicación clara y continua entre todos los miembros del equipo.
- Resolución rápida de problemas: Las dudas y problemas fueron tratados de manera inmediata, evitando retrasos significativos en el progreso del proyecto.

En resumen, el enfoque basado en reuniones semanales y comunicación continua demostró ser efectivo para avanzar en el desarrollo y mejora de la aplicación.

Capítulo 2

Análisis

En esta sección se exploran aspectos clave del proyecto.

Se lleva a cabo un análisis inicial que incluye una evaluación de las aplicaciones actuales, destacando sus defectos y limitaciones. Además, se realiza un estudio de tecnologías para identificar las herramientas más adecuadas para el desarrollo del proyecto.

A continuación, se identifican las oportunidades de mejora en las áreas donde las aplicaciones actuales no cumplen con las necesidades y expectativas de los usuarios. Se presenta una propuesta innovadora que permite a la nueva aplicación mejorar el campo de aplicaciones web para cerraduras inteligentes.

Asimismo, se incluye un análisis de hardware que explica el funcionamiento de las cerraduras VLC y las ventajas de utilizarlas, proporcionando una comprensión integral de cómo estas tecnologías pueden mejorar la seguridad y accesibilidad en diferentes contextos.

2.1. Antecedentes

En el contexto actual de la tecnología de cerraduras inteligentes y gestión de accesos, diversas aplicaciones han emergido para facilitar el control remoto y seguro de accesos a través de dispositivos móviles. Estas soluciones permiten a los usuarios gestionar con conveniencia quién tiene acceso a sus propiedades, mejorando tanto la seguridad como la flexibilidad operativa.

Este proyecto se centra en desarrollar una aplicación innovadora que marque una diferencia significativa en el ámbito de las cerraduras inteligentes. Utilizando tecnologías avanzadas como las cerraduras VLC, la aplicación permitirá generar y compartir claves de manera segura mediante correo electrónico. Esta tecnología no solo simplifica la gestión de

accesos, sino que también aborda preocupaciones de seguridad al eliminar la necesidad de llaves físicas tradicionales.

Al adoptar un enfoque en la mejora continua, la aplicación se propone optimizar la experiencia del usuario al ofrecer una interfaz intuitiva y funcionalidades avanzadas para la gestión eficiente de accesos. Esto incluye la capacidad de administrar permisos de acceso de forma remota y en tiempo real, así como la integración de protocolos de seguridad robustos para proteger la información sensible de los usuarios.

En resumen, este proyecto no solo busca introducir nuevas tecnologías para la gestión de accesos, sino también establecer un estándar superior en términos de seguridad, accesibilidad y comodidad en el control de cerraduras inteligentes en entornos residenciales y comerciales.

2.1.1. Estudio de Apps Similares

Para proporcionar una revisión completa de las aplicaciones de cerraduras inteligentes y gestión de accesos, es crucial no solo resaltar sus características y beneficios, sino también considerar sus limitaciones y posibles defectos:

Estas aplicaciones permiten a los usuarios gestionar el acceso a través de cerraduras inteligentes desde dispositivos móviles, ofreciendo conveniencia y control sobre el acceso físico. Además, todas proporcionan funcionalidades avanzadas como la creación de claves temporales para invitados y la gestión remota de accesos, mejorando la seguridad y flexibilidad para los propietarios.

August Home(1), por ejemplo, facilita la apertura remota de puertas y la gestión de invitaciones de acceso desde dispositivos móviles, lo cual es eficiente para la administración del acceso a propiedades. Sin embargo, algunas críticas señalan posibles vulnerabilidades en la seguridad del sistema de comunicación entre la cerradura y la aplicación. Se puede ver un ejemplo de como es la cerradura en la figura 2.1

Figura 2.1: Cerradura August Wi-Fi Smart

Yale Access (2) ofrece características similares, como la creación de claves temporales y la gestión completa de cerraduras desde la aplicación móvil. A pesar de sus capacidades avanzadas, usuarios han reportado problemas ocasionales con la conectividad Wi-Fi, lo cual puede afectar la confiabilidad de la gestión remota.

Schlage Home (3) permite a los usuarios gestionar códigos de acceso y monitorear el uso de cerraduras en tiempo real, mejorando la seguridad y proporcionando un control detallado sobre el acceso. No obstante, algunos usuarios han mencionado la complejidad en la configuración inicial y la interfaz de usuario que podría ser más intuitiva.

LightKey (4) es una aplicación, que como la que se desarrola, utiliza una cerradura de luz visible (VLC) para abrir puertas como la que se muestra en la figura 2.2. Sin embargo, por la novedad de su tecnología en el area del acceso físico muchos usuarios están encontrando dificultades a la hora de usarla.

Figura 2.2: Cerradura LightKey

2.1.2. Estudio de Tecnologías

Desde el inicio del proyecto se determinó que se desarrollaría una aplicación web en lugar de una aplicación nativa. Esta decisión se basa tanto en la experiencia previa con aplicaciones web como en la falta de disponibilidad de hardware necesario para el desarrollo nativo. Con este punto claro, el siguiente paso crucial fue la selección de las tecnologías a utilizar.

Una aplicación web se divide fundamentalmente en dos componentes principales: el backend y el frontend. El backend es responsable de gestionar la lógica y los datos de la aplicación, mientras que el frontend se encarga de la interfaz de usuario con la cual interactúan los usuarios finales.

Para ambos componentes, se optó por utilizar TypeScript, un lenguaje que ofrece tanto la robustez de la tipificación estática como la flexibilidad del JavaScript moderno.

Backend:

El backend de una aplicación web desempeña un papel fundamental en la gestión de datos y la lógica del negocio. Para este proyecto, se optó por utilizar NestJS(5), un framework de Node.js diseñado para crear aplicaciones escalables y eficientes. NestJS fue seleccionado por su arquitectura modular y su sólido soporte para TypeScript. Además, se integraron GraphQL(6) para una manipulación eficiente de datos y TypeORM(7) como ORM para interactuar de manera efectiva con la base de datos PostgreSQL. PostgreSQL fue la elección debido a su capacidad para manejar cargas de trabajo complejas y su robusta compatibilidad con aplicaciones web a escala empresarial.

Frontend:

El frontend de la aplicación web se encarga de la interfaz de usuario con la cual interactúan directamente los usuarios finales. Se evaluaron tres frameworks destacados:

- **Angular** se encarga de proporcionar una solución completa para el desarrollo frontend. Este framework, mantenido por Google, es conocido por su arquitectura robusta basada en componentes y su capacidad para manejar aplicaciones de gran escala con facilidad.

Angular utiliza TypeScript como su lenguaje principal, lo que facilita el desarrollo de aplicaciones mantenibles y escalables.

- **Vue** es más conocido por su simplicidad y flexibilidad. Este framework progresivo permite a los desarrolladores integrar sus funcionalidades de manera incremental, lo que lo hace ideal para proyectos que requieren una adopción gradual de un framework. Vue es fácil de aprender y ofrece una curva de aprendizaje menos pronunciada en comparación con Angular. Su enfoque en la reactividad y la facilidad de integración con otras bibliotecas o proyectos existentes lo convierte en una opción popular para aplicaciones de tamaño pequeño a mediano. .
- **React** se caracteriza por su enfoque en la construcción de interfaces de usuario a través de componentes reutilizables. Desarrollado y mantenido por Facebook, React utiliza un modelo basado en el DOM virtual(8) que mejora significativamente el rendimiento de las aplicaciones. Este enfoque permite a React actualizar y renderizar eficientemente solo los componentes que han cambiado, lo que resulta en una experiencia de usuario más fluida.

En este proyecto, se optó por React debido a su flexibilidad, rendimiento y la extensa comunidad de desarrollo que lo respalda. La implementación de React utilizando Vite como bundler proporciona un entorno de desarrollo ágil y eficiente, lo que facilita la creación y mantenimiento de la aplicación frontend.

Para la maquetación y los estilos del frontend, se optó por Bootstrap, un framework CSS ampliamente reconocido que ofrece un diseño responsivo y componentes predefinidos. Esta decisión aceleró el desarrollo y garantizó una experiencia de usuario consistente y profesional.

Esta selección meticulosa de tecnologías no solo se fundamentó en la experiencia previa, sino también en la capacidad de cada una para satisfacer los requisitos funcionales y no funcionales del proyecto, asegurando un producto final robusto y escalable.

2.1.3. Brechas y Oportunidades Identificadas

A pesar de las soluciones existentes, se identificaron oportunidades para mejorar la experiencia del usuario y la seguridad en la gestión de accesos mediante una aplicación que genere y comparta claves de cerraduras VLC de manera segura por correo electrónico. Estas oportunidades incluyen:

- Mejorar la accesibilidad y conveniencia para los usuarios autorizados.
- Reducir el riesgo asociado con la pérdida de llaves físicas.
- Proporcionar un sistema más integrado y fácil de usar para la gestión de accesos.

Nuestra aplicación se distingue por enfocarse en la comodidad del usuario, ofreciendo una experiencia intuitiva y accesible. Buscamos facilitar el proceso de gestión de accesos mediante un diseño que prioriza la facilidad de uso, asegurando que los usuarios puedan aprovechar todas las funcionalidades de manera sencilla y eficiente.

2.2. Estado del Arte

Existen diversas situaciones en las que se puede requerir el uso de una cerradura inteligente. Ya sea que se busque una medida más eficiente para acceder al hogar, se hospede en un hotel que utiliza tarjetas de acceso para las habitaciones, o se sea propietario de una vivienda vacacional y no se quiera que se pierda la llave, las cerraduras inteligentes ofrecen una solución ideal. Estas no solo brindan una mayor seguridad, sino que también aportan comodidad y control.

En el ámbito del hospedaje, la gestión de accesos representa un desafío significativo tanto para los propietarios como para los huéspedes. Los propietarios necesitan una manera segura y eficiente de compartir claves de acceso con los huéspedes. Además, en caso de tener más de una alojamiento, se debe poder gestionar múltiples cerraduras y usuarios, manteniendo un control centralizado de todas las propiedades desde una sola plataforma. Esta capacidad de administración centralizada es fundamental para reducir el tiempo y el esfuerzo necesarios para manejar los accesos, especialmente cuando se tienen varios alojamientos.

Por otro lado, los huéspedes requieren un sistema que les permita encontrar y acceder a la propiedad de manera sencilla y sin complicaciones adicionales. La facilidad de uso es un factor crucial para los huéspedes, quienes buscan una experiencia sin fricciones desde el momento en que reservan sus vacaciones. Un sistema de cerraduras inteligentes puede eliminar la necesidad de coordinar la entrega de llaves físicas, permitiendo a los huéspedes acceder al alojamiento mediante códigos temporales o a través de sus teléfonos móviles. Esto no solo mejora la experiencia del

usuario, sino que también incrementa la seguridad al reducir el riesgo de pérdida o copia de llaves físicas.

Además, la implementación de cerraduras inteligentes puede ofrecer beneficios adicionales como el registro de entradas y salidas, lo cual proporciona un nivel adicional de seguridad y control tanto para los propietarios como para los huéspedes. Los propietarios pueden recibir notificaciones en tiempo real sobre quién está accediendo a sus propiedades y en qué momento, lo que les permite responder rápidamente a cualquier situación irregular.

En resumen, las cerraduras inteligentes representan una solución avanzada y efectiva para la gestión de accesos en diferentes contextos, desde la seguridad doméstica hasta la hospitalidad túristica. Al ofrecer una combinación de seguridad, comodidad y control, estas cerraduras satisfacen las necesidades tanto de propietarios como de usuarios finales, facilitando una gestión eficiente y segura de los accesos.

2.2.1. Soluciones Existentes

Actualmente, existen diversas soluciones para la gestión de accesos mediante cerraduras inteligentes. Estas aplicaciones permiten a los usuarios controlar las cerraduras desde dispositivos móviles, ofreciendo ventajas notables:

- Comodidad: Permiten el acceso remoto y la gestión de invitaciones.
- Seguridad: Reducen la dependencia de llaves físicas.

No obstante, también presentan limitaciones que dificultan su adopción y efectividad:

- Complejidad en la Configuración: Muchas aplicaciones requieren configuraciones iniciales complejas.
- Problemas de Conectividad: Dependencia de una conexión constante a Internet.
- Seguridad: Posibles vulnerabilidades en la comunicación entre la cerradura y la aplicación.
- Fragmentación de Funcionalidades: Falta de integración y manejo unificado de múltiples cerraduras y propiedades.

2.2.2. Propuesta de Solución

El proyecto propuesto aborda las deficiencias identificadas en las soluciones existentes mediante una aplicación específica para la gestión de accesos en viviendas vacacionales. Los objetivos principales son:

Facilitar la Gestión de Accesos para Propietarios:

- Generación y Eliminación de Claves: Simplificación del proceso de creación y revocación de claves digitales.
- Gestión de Usuarios y Cerraduras: Facilitar la adición y eliminación de usuarios y la administración de múltiples cerraduras desde una sola plataforma.
- Vista Consolidada: Permitir una vista unificada de todas las propiedades y cerraduras administradas.

Mejorar la Experiencia del Huésped:

- Entrega Segura de Claves: Envío de claves por correo electrónico y acceso a ellas desde la aplicación.
- Localización de Propiedades: Integración de un mapa interactivo para facilitar la ubicación de la propiedad.

Ventajas Diferenciadoras

La aplicación propuesta se diferencia de las soluciones actuales en varios aspectos clave:

- Usabilidad Intuitiva: Diseñada para ser fácil de usar, con interfaces intuitivas tanto para administradores como para huéspedes.
- Integración de Funcionalidades: Consolidación de la gestión de múltiples cerraduras y propiedades en una sola plataforma.
- Seguridad y Eficiencia: Utilización de tecnología VLC para la generación de claves, garantizando comunicaciones seguras y reduciendo el riesgo de pérdidas o errores.
- Experiencia de Usuario Mejorada: Funcionalidades específicas como el envío seguro de claves por correo y mapas interactivos para la localización de propiedades, mejorando significativamente la experiencia del usuario final.

Este proyecto no solo busca mejorar la eficiencia y seguridad de la gestión de accesos para viviendas vacacionales, sino también proporcionar una experiencia de usuario más fluida y satisfactoria, abordando las deficiencias identificadas en las soluciones actuales y ofreciendo una alternativa más integrada y fácil de usar.

2.3. Análisis de Hardware

Este proyecto se centra en el desarrollo de una aplicación para gestionar cerraduras innovadoras basadas en VLC (Visible Light Communication) como la que se muestra en la figura 2.3. A continuación, se explora su funcionamiento y las ventajas que ofrece esta tecnología en comparación con los métodos tradicionales ampliamente conocidos.

Figura 2.3: Cerradura VLC

2.3.1. Funcionamiento

La cerradura VLC se basa en la tecnología de comunicación óptica, donde la luz visible, específicamente emitida desde el LED de la pantalla de un dispositivo móvil, se utiliza como medio para transmitir información de acceso. Cuando un usuario desea desbloquear la cerradura, utiliza la aplicación y esta modula la intensidad de la luz emitida por el LED de la

pantalla. Esta modulación se realiza de manera rápida y precisa, creando variaciones en la luz que codifican un código específico de acceso.

En el extremo receptor, la cerradura está equipada con un sensor óptico, como el que se muestra en la figura2.4, que detecta estas variaciones de luz modulada. Este sensor convierte las señales ópticas en datos digitales comprensibles, decodificando así el código de acceso transmitido desde el dispositivo móvil. Este proceso de decodificación es crucial, ya que permite a la cerradura obtener el código exacto que el usuario ha enviado desde su dispositivo.

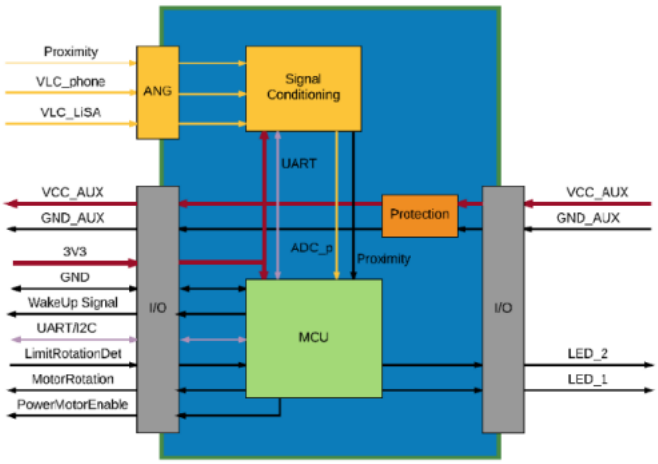


Figura 2.4: Sensor VLC

Una vez que la cerradura ha decodificado el código de acceso, procede a compararlo con los códigos almacenados en su base de datos interna de accesos autorizados. Si el código recibido coincide con uno de los códigos autorizados previamente registrados, la cerradura activa su mecanismo de desbloqueo físico. Esto permite al usuario acceder al área o instalación protegida de manera segura y eficiente.

La cerradura VLC representa una evolución en los sistemas de control de acceso al aprovechar la luz visible como un canal seguro y confiable para la transmisión de datos de acceso. Este enfoque no solo mejora la seguridad, sino que también ofrece una solución tecnológica innovadora adecuada para una variedad de aplicaciones en entornos residenciales, comerciales e industriales donde se requiere un acceso controlado y seguro.

2.3.2. Ventajas

Utilizar tecnología de comunicación por luz visible (VLC) para sistemas de control de acceso ofrece varias ventajas significativas en comparación con métodos como RFID o Bluetooth. Según el estudio "Potentialities and challenges of VLC based outdoor positioning"presentado en la 2015 International Conference on Information Networking (ICOIN), estas ventajas incluyen una alta precisión en la localización y una mayor seguridad debido a la necesidad de línea de visión directa, lo que reduce la posibilidad de interceptación de señales(9). A continuación se presentan varias ventajas:

- **Seguridad Mejorada:** La comunicación VLC utiliza luz visible para transmitir datos, lo cual es altamente seguro debido a que la luz no atraviesa paredes como las ondas de radio. Esto reduce significativamente el riesgo de interceptación no autorizada de datos, mejorando la privacidad y la seguridad del sistema de acceso.
- **Menor Interferencia Electromagnética:** A diferencia de las tecnologías inalámbricas tradicionales como Wi-Fi y Bluetooth, que operan en bandas de frecuencia compartidas y pueden sufrir interferencias, VLC utiliza luz visible, una parte del espectro electromagnético menos congestionada y más predecible. Esto garantiza una comunicación más estable y confiable, especialmente en entornos con alta densidad de dispositivos.
- **Compatibilidad con Dispositivos Existentes:** La mayoría de los dispositivos móviles modernos ya están equipados con los componentes necesarios para la transmisión y recepción de señales VLC, como el LED de la pantalla. Esto permite una implementación rápida y económica del sistema de acceso sin necesidad de hardware adicional costoso o complicado.
- **Eficiencia Energética:** La comunicación VLC consume menos energía en comparación con las tecnologías inalámbricas tradicionales. Los LEDs utilizados para la transmisión son eficientes en términos energéticos y pueden integrarse con sistemas de gestión de energía para optimizar el consumo, lo que resulta en una mayor vida útil de la batería para dispositivos móviles y cerraduras.
- **Aplicaciones Versátiles:** La tecnología VLC es versátil y puede adaptarse a una amplia gama de aplicaciones de control de acceso, incluyendo entornos industriales, comerciales y residenciales. Desde

cerraduras de puertas hasta sistemas de seguridad en grandes instalaciones, VLC proporciona una solución robusta y escalable para gestionar el acceso de manera eficiente y segura.

- **Invulnerabilidad:** Dado que la comunicación se realiza a través de luz visible, es extremadamente difícil interceptar o replicar la señal de manera no autorizada. A diferencia de otros métodos que utilizan señales de radio susceptibles de ser hackeadas, el uso de VLC garantiza un nivel adicional de protección debido a la incapacidad de reproducir la luz visible de manera efectiva, asegurando así la integridad y la seguridad del sistema de acceso.

En conclusión, la adopción de tecnología VLC para sistemas de control de acceso ofrece beneficios significativos en términos de seguridad, eficiencia y compatibilidad.

Capítulo 3

Diseño

El diseño del sistema se centró en las necesidades del usuario, tal como se mencionó en el contexto y justificación del proyecto. Se elaboraron diagramas de casos de uso y flujos de trabajo para visualizar cómo interactuarían los usuarios con la aplicación. Además, se crearon mockups utilizando herramientas como Figma para diseñar la interfaz de usuario y validar la experiencia del usuario antes de la implementación.

3.1. Diagrama de Clases

En el contexto del desarrollo de sistemas de gestión de accesos para viviendas vacacionales, el diagrama de clases juega un papel fundamental en la representación estructural del sistema. Este diagrama ofrece una visión detallada de las clases que componen el sistema, sus atributos, métodos y las relaciones entre ellas.

El diagrama de clases presentado en la siguiente figura 3.1 muestra las clases principales del sistema de gestión de claves, incluyendo las entidades clave como Usuario, Cerradura, Cliente y Administrador. Se destacan las asociaciones entre estas clases, las cardinalidades de las relaciones, y las herencias donde sea aplicable. Este diagrama no solo ilustra la estructura estática del sistema, sino que también sirve como referencia para el desarrollo y la comprensión del código fuente del sistema implementado.

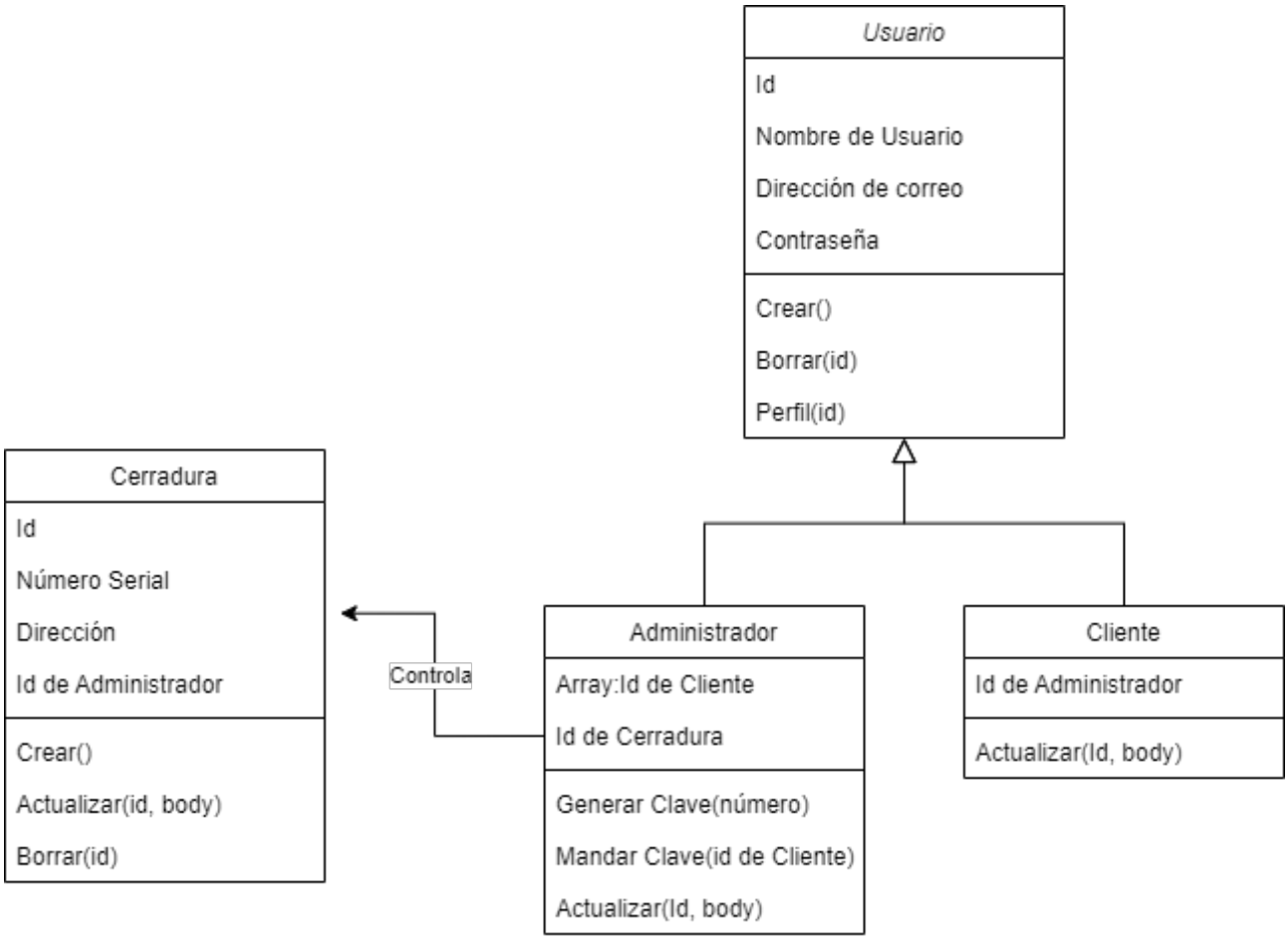


Figura 3.1: Diagrama de Clases

3.1.1. Clase Usuario

Descripción de la Clase:

La clase Usuario representa a los individuos que utilizan el sistema de gestión de accesos para viviendas vacacionales. Almacena información como nombre, dirección de correo electrónico y contraseña.

Atributos:

- id: int - Identificador único del usuario.
- nombre: String - Nombre completo del usuario.
- email: String - Correo electrónico del usuario.
- contraseña: String - Contraseña del usuario.

Métodos:

- Registrar(): Usuario - Devuelve el usuario después de guardar su información en la base de datos.
- Acceder(): Usuario - Devuelve el usuario después de acceder a la aplicación a través de la cuenta creada.
- Borrar(): Usuario - Devuelve el usuario eliminado.
- Perfil(): Usuario - Devuelve el usuario que ejecuta el método.

Relaciones:

La clase Usuario es una clase de la que se hereda por lo que las relaciones más importantes que tiene son con sus hijos: Administrador y Cliente.

3.1.2. Clase Administrador

Descripción de la Clase:

La clase Administrador representa a los usuarios con el poder de organizar las cerraduras y los usuarios que están dentro de la aplicación.

Atributos:

- Clientes: Array de int - Vector con los identificadores de los clientes
- Cerradursa: Array de int - Vector con los identificadores de las cerraduras.

Métodos:

- Actualizar(): Administrador - Devuelve el usuario administrador después de cambiar los atributos requeridos.
- Generar clave(): void - Genera una clave secreta que será la que habra la cerradura.
- Mandar clave(): void - Manda clave generada por correo al usuario que la requiera.

Relaciones:

La clase Administrador tiene una relacion uno a muchos con otras dos clases, Clientes y Cerraduras.Cada administrador puede tener múltiples cerraduras e usuarios para una sola cuenta.

3.1.3. Clase Cliente

La clase Cliente representa a los huespedes que acceden a la clave para abrir la cerradura de la vivienda.

Atributos:

- Admin: int - Número identificativo del administrador que lo controla.

Métodos:

- Actualizar(): Cliente - Devuelve el usuario cliente después de cambiar los atributos requeridos.

3.1.4. Clase Cerradura

La clase Cerradura representa a las cerraduras de la viviendas vacacionales.

Atributos:

- Id: int - Número identificativo de la cerradura.
- Número Serial: string - Secuencia alfanúmerica que identifica a la cerradura física.
- Direccción: string - Dirección real donde la cerradura está instalada.
- Admin: int - Número identificativo del administrador que lo controla.

Métodos:

- Crear(): Cerradura - Devuelve la Cerradura después de crearla y guardarla en la base de datos.
- Borrar(): Cerradura - Devuelve la Cerradura después de borrarla de la base de datos.

- Actualizar(): Cerradura - Devuelve la Cerradura después de cambiar los atributos requeridos.

3.2. Diagrama de Casos de uso

El diagrama de casos de uso es una herramienta fundamental en el diseño de sistemas, ya que permite visualizar las interacciones entre los usuarios (actores) y el sistema. En el contexto de este proyecto, el diagrama de casos de uso se utiliza para capturar y comunicar las funcionalidades clave del sistema de gestión de accesos para viviendas vacacionales.

Los principales objetivos del diagrama de casos de uso en este proyecto son:

- Identificar y representar las interacciones entre los usuarios y el sistema.
- Capturar los requisitos funcionales del sistema de una manera visual y comprensible.
- Facilitar la comunicación entre los desarrolladores y los interesados, asegurando una comprensión común de las funcionalidades del sistema.

El diagrama de casos de uso presentado en la siguiente figura 3.2 incluye los actores principales, como el Cliente, el Administrador y el Sistema de Cerraduras, así como los casos de uso más relevantes, tales como generar clave, mandar clave, y gestionar usuarios.

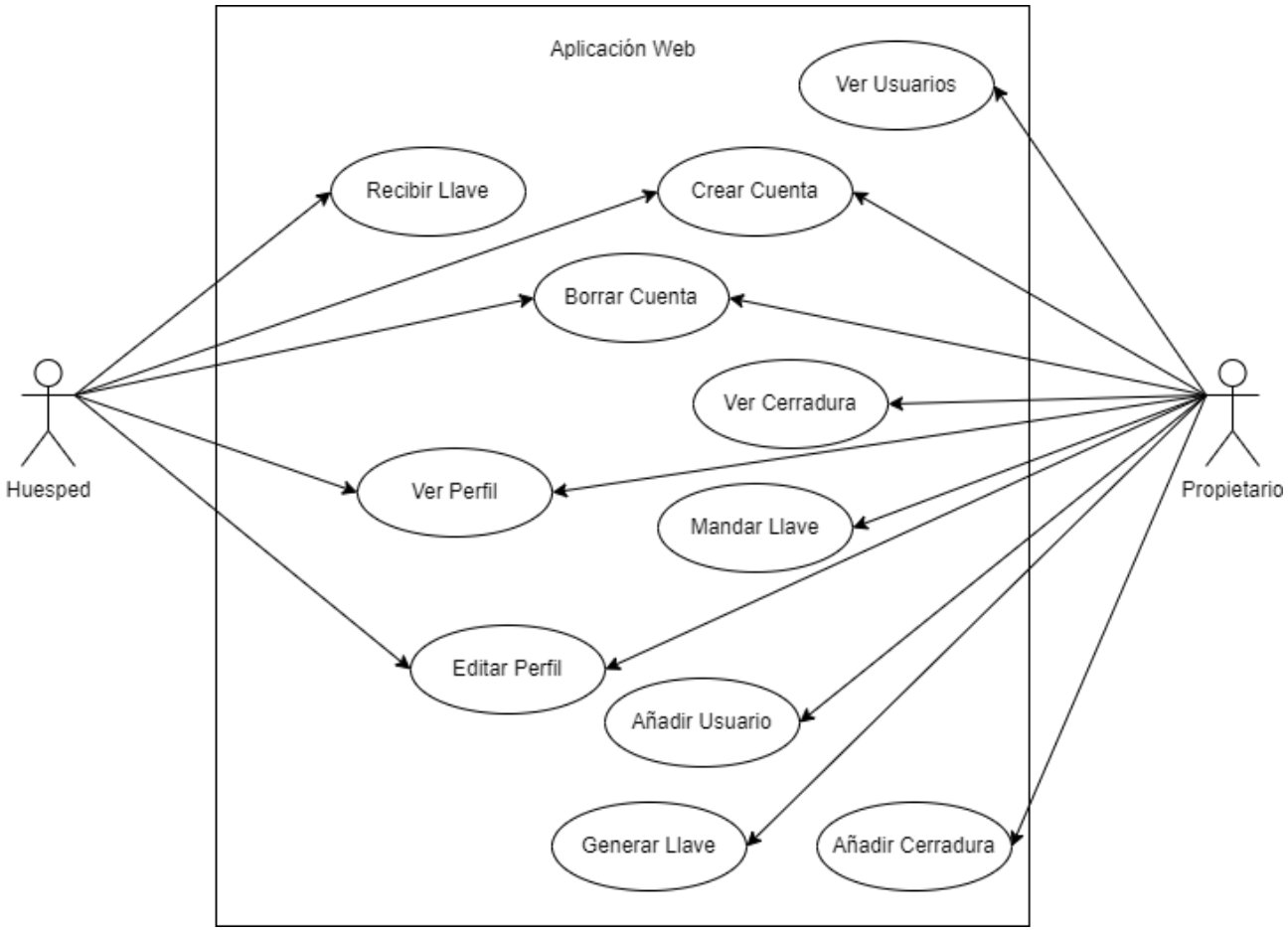


Figura 3.2: Diagrama de Casos de Uso

3.2.1. Actores:

- Huesped(Cliente): Representa a los individuos que utilizan el sistema para obtener las claves de acceso a las viviendas vacacionales.
- Propietario(Administrador): Es el responsable de gestionar el sistema, incluyendo la administración de usuarios y claves.
- Sistema de Cerraduras: Representa el hardware y software de las cerraduras inteligentes que interactúan con las claves generadas.

3.2.2. Casos de Uso:

Generar Clave:

- Descripción: Este caso de uso permite al administrador generar una nueva clave de acceso para una vivienda vacacional.
- Actor Involucrado: Administrador.

- Pasos Principales:
    - El administrador solicita la generación de una nueva clave.
    - El sistema valida la solicitud.
    - Se genera y entrega la clave al usuario.

Mandar Clave:

- Descripción: Este caso de uso permite al administrador mandar una clave por correo electrónico.
- Actor Involucrado: Administrador.
- Pasos Principales:
    - El administrador introduce la clave en el sistema de cerraduras.
    - El sistema verifica el correo electrónico.
    - Se manda según la validación.

Gestionar Usuarios:

- Descripción: Este caso de uso permite al administrador gestionar la información de los usuarios del sistema.
- Actor Involucrado: Administrador.
- Pasos Principales:
    - El administrador accede al panel de gestión.
    - Realiza operaciones como añadir, modificar o eliminar usuarios.
    - El sistema actualiza la información de los usuarios.

Gestionar Cerraduras:

- Descripción: Este caso de uso permite al administrador gestionar la información de las cerraduras del sistema.
- Actor Involucrado: Administrador.
- Pasos Principales:
    - El administrador accede al panel de gestión.
    - Realiza operaciones como añadir, modificar o eliminar cerraduras.
    - El sistema actualiza la información de las cerraduras.

Gestionar Cuenta:

- Descripción: Este caso de uso permite tanto al cliente como al administrador gestionar su cuenta.
- Actor Involucrado: Administrador/Cliente.
- Pasos Principales:
  - El administrador/usuario crea o borra su cuenta.
  - El sistema actualiza la información de los usuarios.

Gestionar Perfil:

- Este caso de uso permite tanto al cliente como al administrador gestionar su perfil.
- Actor Involucrado: Administrador/Cliente.
- Pasos Principales:
  - El administrador accede al perfil.
  - Realiza operaciones como modificar su información y la guarda.
  - El sistema actualiza la información de los usuarios.

3.3. Diagrama de flujos

El diagrama de flujo es una representación visual que muestra la secuencia de pasos y decisiones dentro de un proceso o funcionalidad específica del sistema. Es útil para clarificar la lógica detrás de los flujos de trabajo y para identificar puntos críticos en el proceso.

Los principales objetivos del diagrama de flujo en este proyecto son:

- Visualizar claramente las etapas y decisiones dentro del proceso.
- Identificar posibles mejoras en la eficiencia del proceso.
- Facilitar la comprensión y comunicación de la lógica del sistema a diferentes partes interesadas.

El siguiente diagrama, representado en la figura3.3 ilustra el proceso de generación de claves de acceso para viviendas vacacionales dentro del sistema desarrollado.

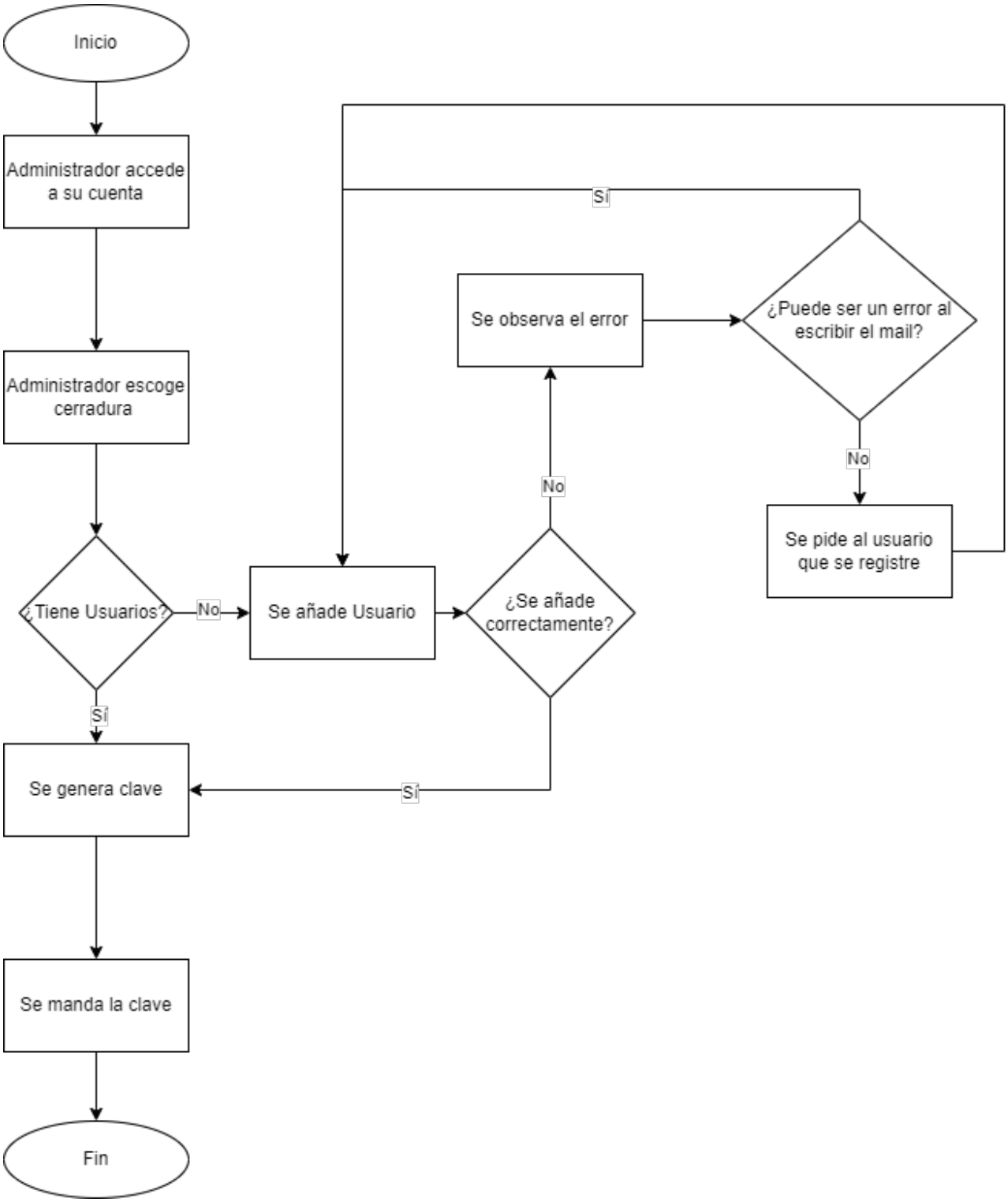


Figura 3.3: Flujo de la generación de claves

3.4. MockUps

Los mockups, o maquetas visuales, desempeñan un papel crucial en el desarrollo y diseño de cualquier proyecto de software o aplicación. Estas

representaciones gráficas ofrecen una vista preliminar de cómo se verá y funcionará la interfaz de usuario final, proporcionando una representación tangible de las ideas y conceptos abstractos que se han ideado durante la fase de diseño. En el contexto de este proyecto, los mockups han sido desarrollados como herramienta fundamental para visualizar y refinar la experiencia del usuario.

3.4.1. Mockup SignIn:

- Descripción: La figura 3.4 representa el mockup de la página de inicio de sesión estándar de la aplicación. Está diseñado con colores blanco, gris, negro y verde, que son los colores de la empresa de cerraduras para la cual está inspirada la app.
- Elementos Clave: Incluye un cuadro central blanco con campos de entrada para el nombre de usuario y contraseña, diseñado para una identificación rápida y accesibilidad.

Figura 3.4: MockUp Sign In

3.4.2. Mockup SignUp:

- Descripción: Similar al SignIn, la figura 3.5 representa el mockup de la página de registro para nuevos usuarios.
- Elementos Clave: Mantiene la misma estructura visual con campos de entrada para información de registro como nombre, correo electrónico y contraseña, adaptados al esquema de colores de la empresa.

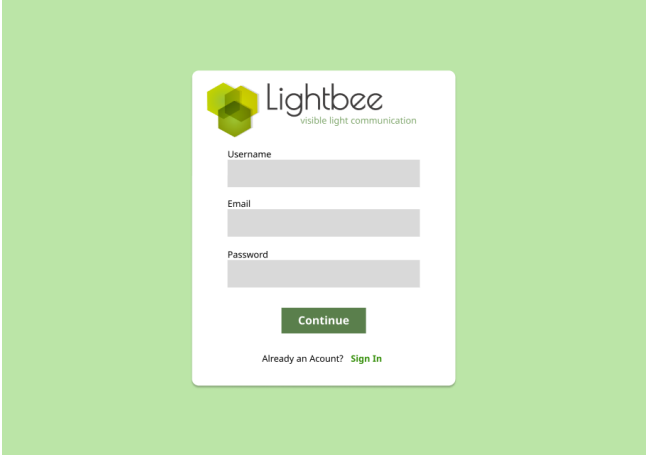


Figura 3.5: MockUp Sign Up

3.4.3. Mockup SignUp Serial Number:

- Descripción: La figura 3.6 respresenta el mockup de la página que es la continuación del proceso de registro (SignUp), específicamente para administradores. Permite ingresar el número serial de la cerradura que desean gestionar.
- Elementos Clave: Incorpora los campos necesarios para que los administradores introduzcan el número serial de la cerradura, mientras que los usuarios huéspedes deben dejar este campo en blanco.

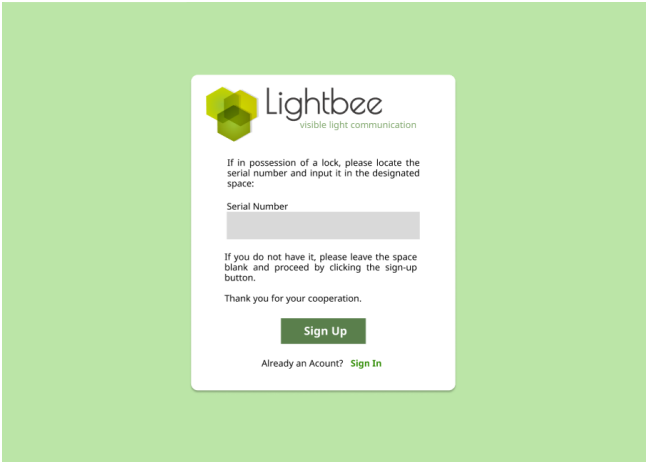


Figura 3.6: MockUp Sign Up/Serial Number

3.4.4. Mockup Página Principal Admin:

- Descripción: La figura 3.7 representa el mockup de la página principal para administradores, ofreciendo una visión general de todas las

cerraduras bajo su administración y los usuarios asociados.

- Elementos Clave: Muestra un listado de cerraduras con detalles como usuarios activos, permitiendo a los administradores interactuar con estos elementos para gestionar accesos.

Figura 3.7: MockUp Página principal para Administradores

3.4.5. Mockup Página Principal Client:

- Descripción: Para usuarios regulares, La figura 3.8 representa el mockup de la página principal muestra la cerradura a la cual tienen acceso. Proporciona detalles como nombre del sitio, dirección, número de contacto del administrador de la llave y un mapa interactivo para la ubicación exacta.
- Elementos Clave: Facilita una experiencia centrada en la cerradura específica del usuario, asegurando que la información clave sea accesible de manera intuitiva.

Figura 3.8: MockUp Página Principal para Clientes

3.4.6. Mockup Perfil:

- Descripción: La figura 3.9 representa el mockup de la página de perfil del usuario, accesible para cualquier tipo de usuario registrado.
- Elementos Clave: Permite al usuario modificar sus datos personales, garantizando una gestión fácil y segura de la información del perfil.

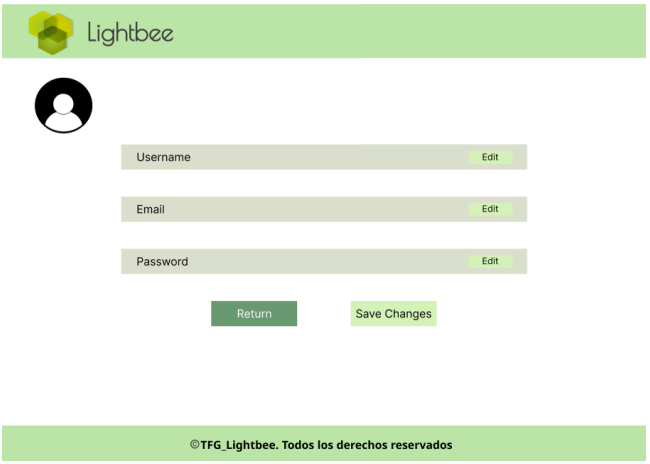


Figura 3.9: MockUp Perfíl de usuarios

3.4.7. Mockup Añadir Cerradura:

- Descripción: En la figura 3.10 que representa al mockup, los administradores pueden buscar y añadir nuevas cerraduras a su cuenta principal.

- Elementos Clave: Incluye un campo de búsqueda por número serial para localizar rápidamente la cerradura deseada y agregarla a su listado administrativo.

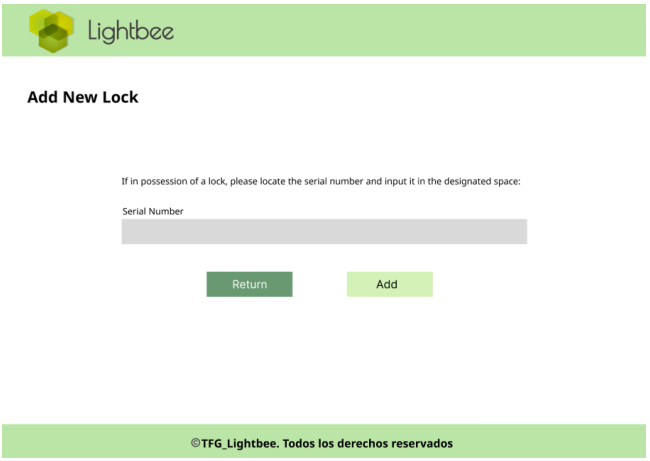


Figura 3.10: MockUp Página para añadir cerraduras

3.4.8. Mockup Añadir Usuario:

- Descripción: Diseñado para administradores, la figura 3.11 representa un mockup que permite buscar usuarios por correo electrónico y añadirlos a una cerradura específica.
- Elementos Clave: Ofrece funcionalidades de búsqueda y gestión de usuarios, facilitando la asignación de accesos a distintas cerraduras según sea necesario.

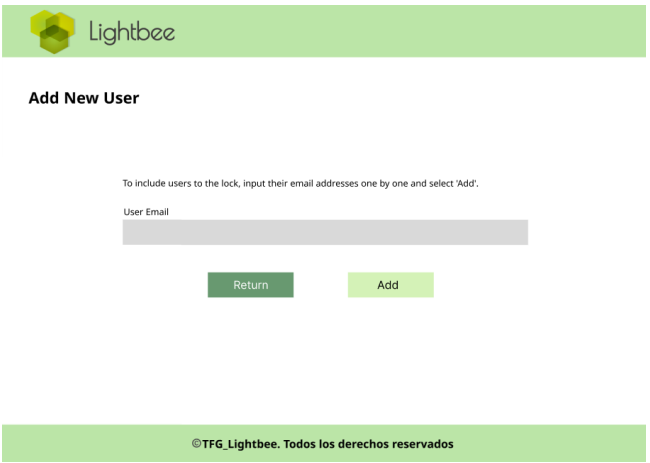


Figura 3.11: MockUp Página para añadir usuarios

Cada mockup está alineado con los colores corporativos y diseñado para proporcionar una experiencia de usuario fluida y coherente, enfocándose en la funcionalidad intuitiva y la usabilidad práctica para todos los tipos de usuarios de la aplicación.

3.5. Diseño de la Base de Datos

El diseño de la base de datos en este proyecto utiliza PostgreSQL para almacenar y gestionar eficientemente la información relacionada con las cerraduras VLC, administradores y clientes. Aquí se detallan los aspectos esenciales del diseño:

3.5.1. Modelo de Datos Relacional

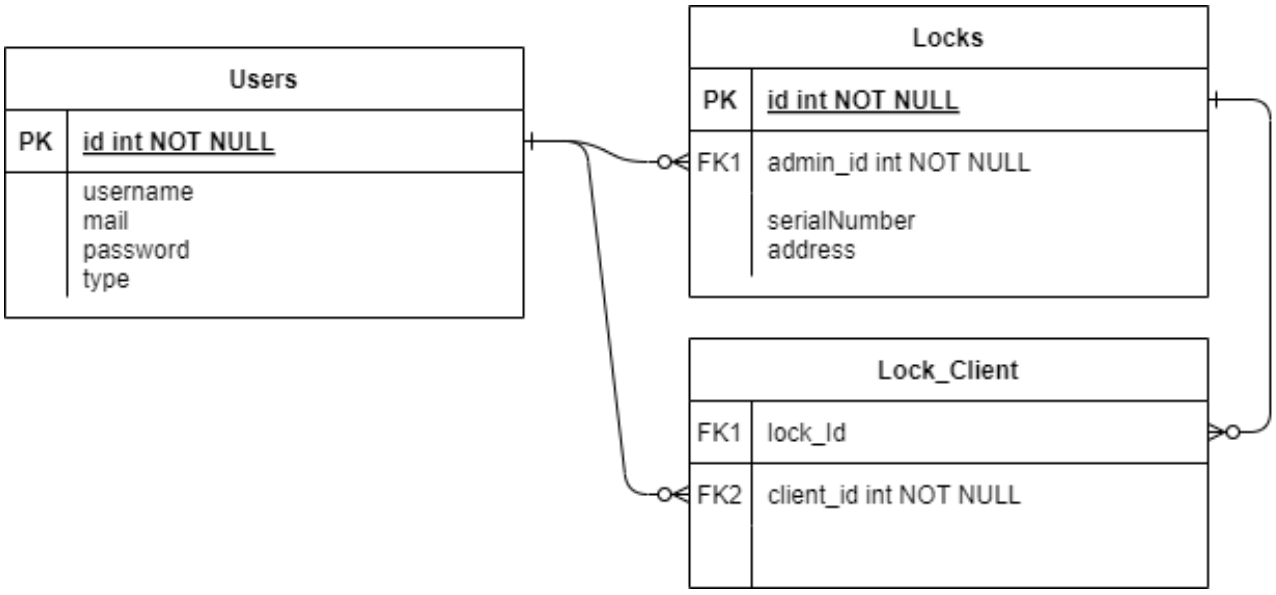


Figura 3.12: Esquema del Modelo de Datos

3.5.2. Esquema de la Base de Datos

El esquema de la base de datos presentado por la figura 3.12 se organiza en tres tablas principales: Users, Lock y Lock-Client, las cuales almacenan información y gestionan las relaciones entre usuarios y cerraduras.

La tabla Users almacena información común a todos los usuarios del sistema. Sus atributos principales incluyen id (clave primaria), username, password, entre otros datos relevantes para la identificación y autenticación de los usuarios.

La tabla Lock representa las cerraduras VLC gestionadas por los administradores. Sus atributos incluyen id (clave primaria), serialNumber, address y adminId (clave foránea que referencia a un administrador en la tabla Users). La relación entre la tabla Lock y la tabla Users es de muchos a uno, ya que un administrador puede gestionar múltiples cerraduras.

Finalmente, la tabla de relación Lock-Client implementa la relación muchos a muchos entre las cerraduras (Lock) y los clientes (Client). Esta tabla contiene los atributos lockId (clave foránea hacia la tabla Lock) y clientId (clave foránea hacia la tabla Users, específicamente para los clientes). Esta estructura permite que una cerradura pueda ser asignada a múltiples clientes y viceversa, facilitando una gestión eficiente y flexible de las relaciones entre usuarios y cerraduras.

3.5.3. Normalización y Optimización

Se aplicaron principios de normalización para reducir la redundancia como el principio de Simplicidad y mejorar la integridad de los datos. Además, se optimizaron las consultas mediante el uso de índices en las claves foráneas y otras columnas clave para mejorar el rendimiento de las consultas.

3.5.4. Seguridad y Privacidad

Se implementaron medidas de seguridad como el cifrado de contraseñas (password) y la gestión adecuada de permisos de acceso para proteger la información sensible almacenada en la base de datos.

3.5.5. Escalabilidad y Mantenimiento

El diseño está preparado para escalar de manera eficiente, permitiendo la adición de nuevos administradores (Admin), clientes (Client) y cerraduras (Lock) según las necesidades del sistema. Se establecieron procedimientos para el mantenimiento regular de la base de datos y la optimización de su rendimiento.

3.5.6. Integración con el Backend

La estructura de la base de datos está diseñada para integrarse estrechamente con el backend del sistema, facilitando la manipulación eficiente

de datos a través de consultas SQL optimizadas y procedimientos almacenados.

El diseño ampliado de la base de datos proporciona una estructura robusta y eficiente para almacenar y gestionar información relacionada con los usuarios, administradores, clientes y cerraduras en el sistema. Está diseñado para cumplir con requisitos de rendimiento, seguridad y escalabilidad, asegurando una gestión eficiente de los datos del sistema y una experiencia óptima para los usuarios finales.

Capítulo 4

Implementación

4.1. Desarrollo

El desarrollo del proyecto se dividió en tres fases principales: Interfaz Web, Backend y Frontend. Cada fase se llevó a cabo con un enfoque particular en asegurar la funcionalidad, seguridad y usabilidad del sistema.

4.1.1. Interfaz Web

En esta fase, el objetivo principal fue desarrollar el HTML necesario para generar y transmitir la clave visible al sensor de luz. Se realizaron investigaciones detalladas sobre la modulación y funcionamiento de cerraduras similares, apoyándose en artículos científicos como "Low Power Control Access System based on VLC for Industrial Applications"presentado en la IECON 2022(10). Estos estudios fueron fundamentales para adquirir un conocimiento profundo de la tecnología, facilitando así su implementación.

Además, se evaluaron diversas tecnologías y librerías JavaScript para garantizar la reproducción precisa de la clave en HTML. Se adoptó un enfoque de prueba y error, desarrollando varios prototipos para evaluar la visibilidad y funcionalidad de las claves generadas. Esto implicó ajustes en el diseño e implementación de múltiples iteraciones hasta alcanzar un resultado óptimo.

Las herramientas utilizadas durante el desarrollo fueron:

- HTML5: Para la estructura de la página.
- CSS3: Para el diseño y estilo visual.
- JavaScript: Para la funcionalidad interactiva.

La fase concluyó con éxito cuando el código HTML no solo funcionaba correctamente, sino que también se logró transmitirlo a través de Gmail

mediante un enlace, garantizando su uso práctico y accesibilidad.

4.1.2. Backend

El desarrollo del backend se centró en establecer las funcionalidades esenciales de la aplicación, asegurando una gestión eficiente y segura de las claves. Se diseñó una arquitectura robusta utilizando TypeScript y NestJS, seleccionados por su capacidad para mantener un código modular y escalable.

Se implementó la lógica de negocio necesaria utilizando TypeORM junto con PostgreSQL para la manipulación segura de datos críticos como la generación, almacenamiento, validación y revocación de claves. La integración con PostgreSQL garantizó un almacenamiento seguro y escalable de la información.

Además, se optó por GraphQL para facilitar la consulta eficiente de datos y optimizar las interacciones entre el frontend y el backend. Esto permitió una gestión más flexible de las operaciones CRUD (Create, Read, Update, Delete) relacionadas con las claves y otros recursos del sistema.

Se implementó también un sistema de seguridad basado en JSON Web Tokens (JWT) para autenticar usuarios y autorizar el acceso a recursos protegidos, mejorando así la seguridad del sistema y la gestión de sesiones.

Al concluir esta fase, se estableció un backend completo y bien estructurado que facilita de manera efectiva la gestión segura y eficiente de claves, asegurando la confidencialidad y la integridad de los datos sensibles de los usuarios. Esta implementación integral no solo mejoró la seguridad y el rendimiento del sistema, sino que también sentó las bases para futuras expansiones y mejoras en la aplicación.

4.1.3. Frontend

En esta fase, se desarrolló una interfaz de usuario intuitiva y amigable para facilitar la gestión de claves de acceso. Se comenzó con estudios detallados de usabilidad y diseño de wireframes, asegurando una estructura que optimizara la navegación y la interacción del usuario.

La implementación se realizó utilizando tecnologías modernas como React.js, que permitieron la creación de componentes modulares y eficientes. Se realizaron pruebas exhaustivas con usuarios finales para validar la usabilidad y se utilizó Redux para gestionar el estado global de la aplicación de manera coherente.

Como resultado de este enfoque integral, se logró desarrollar una interfaz de usuario que facilita a los usuarios la gestión intuitiva de sus claves de acceso, asegurando una experiencia fluida y satisfactoria en el uso cotidiano del sistema.

4.2. Despliegue

El objetivo principal en esta fase fue garantizar la disponibilidad y el rendimiento óptimo del sistema a través de una plataforma de despliegue adecuada. Tras una evaluación exhaustiva de diversas opciones, elegimos utilizar Netlify(11) por su facilidad de uso y su capacidad para alojar aplicaciones web de manera gratuita en el frontend. Además, para el backend, optamos por Render(12) debido a su capacidad para escalar automáticamente las aplicaciones, manejar cargas de trabajo intensivas de manera eficiente y proporcionar un entorno robusto y confiable para ejecutar nuestro servicio. Esta combinación nos permitió implementar una arquitectura completa que garantiza tanto la estabilidad como la escalabilidad de nuestro sistema.

Se procedió configurando meticulosamente el entorno de producción para optimizar el rendimiento y garantizar la seguridad del sistema. Esto incluyó la configuración de variables de entorno, ajustes de escalabilidad y configuración de seguridad.

Durante todo el proceso de despliegue, se emplearon prácticas de integración continua y entrega continua (CI/CD) utilizando herramientas como Git y pipelines automatizados. Esto permitió realizar actualizaciones eficientes y consistentes del sistema, garantizando la estabilidad y la disponibilidad continua para los usuarios finales.

Como resultado de estas medidas, se logró establecer un entorno de producción robusto en Netlify, proporcionando una base segura y escalable para la aplicación. Esto no solo aseguró una experiencia de usuario sin interrupciones, sino que también sentó las bases para futuras expansiones y mejoras del sistema.

4.3. Seguridad y Privacidad

En este proyecto, la seguridad y la privacidad de los datos de los usuarios son aspectos fundamentales que se han considerado desde el inicio del desarrollo. A continuación, se detallan las medidas de seguridad im-

plementadas para proteger la información de los usuarios y asegurar su privacidad.

4.3.1. Cifrado de Datos

El objetivo principal del proyecto fue proteger la información sensible de los usuarios tanto durante su almacenamiento como en su transmisión, asegurando que solo las partes autorizadas puedan acceder a los datos.

En cuanto a la implementación de medidas de seguridad:

Para cifrar los datos durante su transmisión, se implementó el protocolo HTTPS. Este asegura que la información intercambiada entre el cliente y el servidor esté protegida mediante cifrado, evitando así que pueda ser interceptada o modificada por terceros.

Para almacenar de manera segura las contraseñas de los usuarios en la base de datos, se utilizó bcrypt. Este algoritmo de hashing adaptativo aplica técnicas que dificultan ataques de fuerza bruta y garantiza la seguridad de las contraseñas almacenadas, incluso en caso de que la base de datos sea comprometida.

Estas medidas de seguridad fueron fundamentales para cumplir con los estándares de protección de datos y asegurar la confidencialidad e integridad de la información de los usuarios en todo momento.

4.3.2. Autenticación y Autorización

Para asegurar que solo usuarios autenticados y autorizados puedan acceder a los recursos y funcionalidades del sistema, se implementaron diversas medidas clave. Se estableció un sistema de autenticación basado en JSON Web Tokens (JWT). Cada usuario que se autentica recibe un token único que se utiliza para verificar su identidad en cada solicitud, garantizando un acceso seguro a los recursos protegidos del sistema.

4.3.3. Protección contra Ataques Comunes

Para asegurar la integridad y privacidad de los datos del sistema, se implementaron diversas medidas contra ataques comunes. En primer lugar, se utilizó consultas preparadas y ORM al interactuar con la base de datos para prevenir Inyecciones SQL. Esta estrategia asegura que todas las consultas sean estructuradas de manera segura, eliminando la posibilidad de que se introduzcan comandos maliciosos a través de entradas de usuario.

Capítulo 5

Resultados

5.1. Diferencias y Evolución del Diseño

Durante el proceso de desarrollo, se implementaron diversas modificaciones significativas en el diseño original:

- **Ajuste de la Paleta de Colores:** Se realizó un ajuste en la paleta de colores para mejorar la legibilidad y la coherencia visual como se muestra en la figura 5.1. Debido a que el logo de la empresa presenta letras transparentes que no eran visibles con el fondo verde del navegador, se decidió cambiar el color del navegador al blanco del fondo de las páginas. Esto aseguró que el contenido del navegador fuera claramente visible para los usuarios.

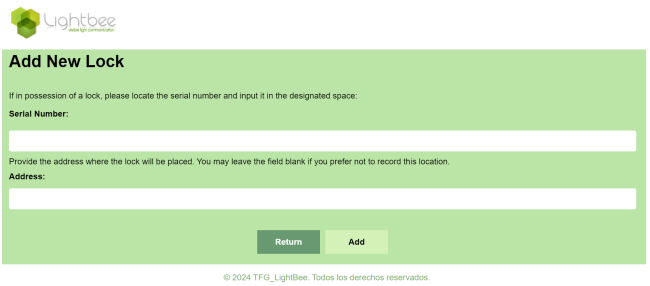


Figura 5.1: Página para añadir cerraduras

- **Actualización de la Iconografía:** Se optó por utilizar iconos más reconocibles y convencionales. Además, se añadieron nuevos iconos para funciones adicionales que surgieron durante el proceso de prototipado, como la capacidad de eliminar cerraduras. Estos cambios no solo mejoraron la estética visual, sino que también facilitaron la comprensión y la navegación para los usuarios.
- **Introducción de Nuevas Páginas:** Se añadieron nuevas páginas

para mejorar la navegación y la interacción del usuario. Por ejemplo, en la página de SignUp, que es la que se muestra en la figura 5.2, se modificó el flujo original para incluir una pregunta inicial sobre la posesión de una cerradura. Esto simplificó el proceso de registro y clarificó la acción que el usuario estaba realizando en cada paso.

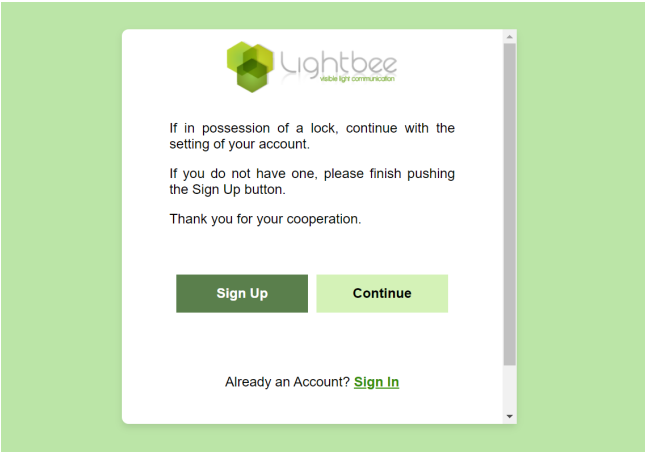


Figura 5.2: Página entre el registro y la adición de la cerradura

- **Adición de Elementos en Páginas:** Se identificó la falta de elementos necesarios en ciertas páginas originales para la funcionalidad completa. Por ejemplo, en la página de añadir cerradura se añadió la necesidad de ingresar tanto el número serial como la dirección donde se encuentra la cerradura como se muestra en la figura 5.1. Esta adición aseguró que los usuarios pudieran completar la acción de manera efectiva y sin inconvenientes.
- **Refinamiento en la Interacción con Elementos:** Se simplificó la forma en que los usuarios interactúan con ciertos elementos para mejorar la usabilidad. Por ejemplo, en la página de addUser se implementó un buscador como el que se muestra en la figura 5.3. Este sirve para encontrar usuarios existentes utilizando tanto el correo electrónico como el nombre de usuario. Este cambio facilitó la búsqueda y selección de usuarios, mejorando la eficiencia y la experiencia del usuario.

Figura 5.3: Página para añadir usuarios con buscador

Estas modificaciones fueron fundamentales para optimizar el diseño y asegurar que el producto final no solo cumpliera con los requisitos funcionales, sino que también ofreciera una experiencia de usuario intuitiva y satisfactoria.

5.2. Aspectos de Usabilidad y Experiencia del Usuario

Durante el desarrollo del frontend, se hizo un esfuerzo por asegurar que la interfaz fuera accesible y fácil de usar para todos los usuarios. Aunque se logró un diseño responsivo, se reconoce la necesidad de mejorar la accesibilidad con teclado, especialmente en áreas críticas como la página de añadir usuarios, donde actualmente existen limitaciones.

Además, se seleccionó cuidadosamente una gama de colores que optimice la legibilidad. Se utilizarón herramientas de contraste para asegurar que los colores cumplieran con las pautas de accesibilidad.

Para facilitar la interacción, se utilizaron iconografías claras como el ícono de ’más’ para añadir y el ícono de un sobre para enviar por email. Estos iconos fueron validados con pruebas de usuario para asegurar que fueran intuitivos y comprensibles.

Nuestro proceso de diseño ha sido iterativo, con múltiples ajustes basados en el feedback de los usuarios y pruebas de usabilidad. Esto nos ha permitido mejorar continuamente la disposición de los elementos y la funcionalidad general de la interfaz para asegurar una experiencia fluida y satisfactoria para todos los usuarios.

5.3. Gestión de Avisos

Durante el desarrollo de la interfaz de usuario, implementamos un sistema de avisos emergentes diseñado para informar de manera clara y concisa

al usuario sobre el resultado de sus acciones. Estos avisos se dividen en dos tipos principales: errores y éxitos, cada uno destinado a situaciones específicas para mantener al usuario informado en todo momento.

5.3.1. Avisos de Error

Los avisos de error como el que se presenta en la figura 5.4,aparecen cuando se produce una situación inesperada o incorrecta durante la interacción del usuario con la aplicación. Estos avisos están diseñados para alertar al usuario sobre problemas como entradas inválidas, acciones no permitidas o cualquier otro escenario que requiera atención inmediata.

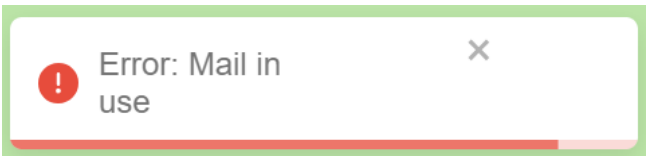


Figura 5.4: Aviso de Error dentro de la aplicación

5.3.2. Avisos de Éxito

Por otro lado, los avisos de éxito como el que se presenta en la figura 5.5,se muestran para confirmar que una acción se ha completado con éxito. Estos avisos proporcionan feedback positivo al usuario, asegurando que estén informados cuando una tarea o proceso se haya realizado correctamente.

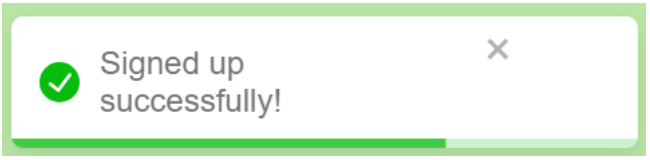


Figura 5.5: Aviso de Éxito dentro de la aplicación

5.3.3. Implementación y Uso

Cada tipo de aviso ha sido diseñado con un estilo visual distintivo para garantizar que sean fácilmente reconocibles. Utilizamos colores y símbolos específicos que reflejan claramente la naturaleza del aviso: rojo para errores y verde para éxitos. Además, se incorporan iconografías adicionales

para mejorar la comprensión rápida del mensaje del aviso, como un símbolo de stop para los errores y un tick para los éxitos.

Además de su diseño visual, los avisos ofrecen dos opciones de interacción para mejorar la experiencia del usuario:

- Cierre Manual: Los usuarios pueden cerrar el aviso antes de que se complete el tiempo de visualización haciendo clic en la ’x’ en la esquina superior derecha del aviso.
- Tiempo de Visualización Extendido: Al mantener el ratón sobre el aviso, se detiene el temporizador, permitiendo a los usuarios leer el mensaje con tranquilidad antes de que desaparezca automáticamente.

Estas opciones están diseñadas para facilitar una interacción intuitiva y efectiva con los avisos, asegurando que los usuarios puedan gestionar fácilmente la información proporcionada sin interrupciones innecesarias.

5.4. Pruebas y Validación

Las pruebas son procedimientos críticos en el desarrollo de software y hardware que garantizan su funcionamiento correcto y confiabilidad. Las pruebas unitarias se centran en verificar el comportamiento de componentes individuales de software, mientras que las pruebas de hardware evalúan la funcionalidad física y electrónica de dispositivos y circuitos. Ambos tipos de pruebas son fundamentales para detectar errores, optimizar el rendimiento y asegurar la calidad antes de la implementación y uso final del proyecto.

5.4.1. Pruebas Unitarias

Durante el desarrollo del proyecto, se priorizó la implementación de pruebas unitarias como parte integral del proceso de aseguramiento de la calidad del software. Estas pruebas fueron diseñadas para validar la funcionalidad individual de cada componente y función dentro del sistema, garantizando su correcto funcionamiento y comportamiento esperado.

Herramientas Utilizadas

Se empleó Jest como la herramienta principal para la ejecución de las pruebas unitarias. Jest es conocido por su simplicidad y eficacia en la escritura y ejecución de pruebas en aplicaciones JavaScript y TypeScript.

Además, su capacidad para realizar pruebas de manera paralela y proporcionar informes detallados facilitó la identificación temprana de errores y problemas potenciales en el código.

Objetivos de las Pruebas Unitarias

Las pruebas unitarias se centraron en los siguientes objetivos clave:

- Validación de Funcionalidades Críticas: Cada función y método crucial en el sistema fue sometido a pruebas exhaustivas para verificar su comportamiento bajo diversas condiciones y entradas.
- Identificación de Errores de Lógica: Se buscó detectar y corregir errores de lógica antes de la integración con otras partes del sistema, minimizando así el impacto de los errores en etapas posteriores del desarrollo.
- Garantía de Estabilidad y Fiabilidad: Asegurar que las actualizaciones y cambios en el código no afectaran negativamente la estabilidad del sistema existente, manteniendo la confianza en su funcionamiento en producción.

Metodología de Pruebas

Las pruebas unitarias fueron desarrolladas siguiendo una metodología estructurada:

- Setup y Tear Down: Se establecieron configuraciones iniciales y limpieza de estados después de cada prueba para mantener un entorno de prueba consistente y predecible.
- Mocks y Stubs: Se utilizaron mocks y stubs para simular el comportamiento de dependencias externas y facilitar la prueba aislada de componentes individuales.
- Assertions Detalladas: Cada prueba incluyó assertions detalladas para verificar los resultados esperados y asegurar que la función o método probado cumpliera con los requisitos de negocio y técnicos.

Cobertura

La implementación de pruebas unitarias permitió alcanzar una cobertura media de código como muestra la figura 5.6, asegurando que cierta

parte de las funcionalidades críticas estuvieran protegidas por pruebas automatizadas.

File	% Stmts	% Branch	% Funcs	% Lines
All files	69.84	58.82	39.42	67.24
lock	61	44.44	39.47	59.78
lock.entity.ts	58.33	100	0	57.14
lock.resolver.ts	77.41	100	56.25	75.86
lock.service.ts	51.11	44.44	50	50
lock/dto	70.83	100	0	65
createLock.dto.ts	72.72	100	0	66.66
updateLock.dto.ts	69.23	100	0	63.63
mail	71.42	100	0	60
mail.service.ts	71.42	100	0	60
user/admin	71.42	50	50	70.12
admin.entity.ts	69.23	100	0	70
admin.resolver.ts	84.61	100	57.14	83.78
admin.service.ts	56.25	50	62.5	53.33
user/admin/dto	87.5	100	0	83.33
updateAdmin.dto.ts	87.5	100	0	83.33
user/client	75.51	100	56.25	73.8
client.entity.ts	100	100	100	100
client.resolver.ts	79.16	100	54.54	80.95
client.service.ts	63.15	100	60	58.82
user/client/dto	100	100	100	100
updateClient.dto.ts	100	100	100	100
user/user	72.97	100	36.36	69.69
user.entity.ts	62.5	100	0	57.14
user.service.ts	80.95	100	80	78.94
user/user/dto	72.72	100	0	66.66
updateUser.dto.ts	72.72	100	0	66.66

Figura 5.6: Resultado de jest-coverage

5.4.2. Pruebas de Hardware

También se realizaron pruebas de hardware utilizando los componentes presentados en la figura 5.7. Este conjunto incluye un sensor de luz visible conectado a una placa electrónica, que a su vez está conectada a una cerradura. Un portapilas proporciona la alimentación necesaria y un cable adaptador USB a serial permite la conexión entre la placa y un ordenador, facilitando la comunicación a través de Docklight Scripting(13). El script utilizado para estas pruebas fue proporcionado por el cotutor.

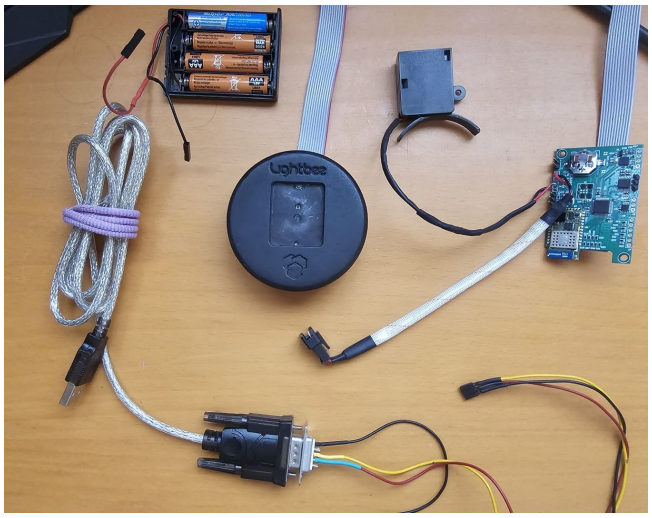


Figura 5.7: Elementos físicos necesarios para las pruebas

El procedimiento de pruebas se realizó de la siguiente manera:

1. Configuración del Sistema

   a) Conecta el portapilas a la placa electrónica para suministrar energía al sistema.

   b) Conecta el cable adaptador USB a serial a la placa y al ordenador, asegurando una comunicación adecuada con Docklight.

   c) Abre el script proporcionado en Docklight.

2. Activación del Sensor

   a) Activa el sensor de luz visible colocando un objeto delante de él. La activación del sensor se indica mediante una luz naranja parpadeante.

   b) Espera hasta que el sistema esté completamente operativo y el sensor haya sido activado correctamente.

3. Prueba de la Clave

   a) Una vez que el sensor está activo, prueba la clave de luz visible sobre el sensor.

   b) Si la clave es correcta, el sensor cambiará de color a verde y en la pantalla de Docklight se mostrará como en la figura5.8 confirmando la validez de la clave.

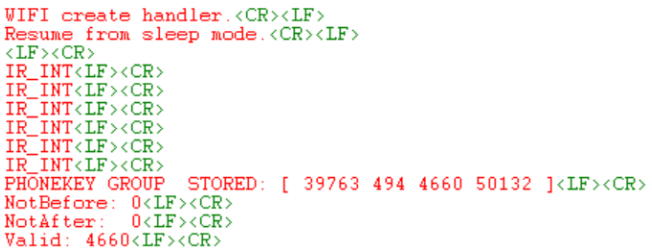


Figura 5.8: Resultado de DockLight cuando la clave es la correcta

c) Si la clave es incorrecta, el sensor se iluminará en rojo y se mostrará como en la figura 5.9 en la pantalla indicando el error.

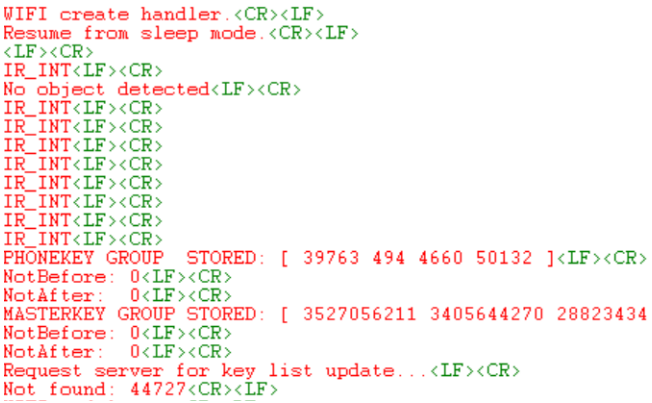


Figura 5.9: Resultado de DockLight cuando la clave no es correcta

d) Si no se detecta ningún objeto, Docklight también notificará esta situación.

Estas pruebas aseguran que la transmisión y recepción de claves a través del sensor de luz visible sean precisas y confiables. La integración de Docklight para la monitorización y el análisis en tiempo real permite identificar y corregir rápidamente cualquier discrepancia, garantizando así la seguridad y funcionalidad del sistema de cerraduras electrónicas.

Capítulo 6

Conclusiones y Líneas Futuras

6.1. Conclusiones

Durante el desarrollo de este proyecto, se alcanzaron con éxito los objetivos fundamentales que se propusieron inicialmente. La aplicación diseñada ha hecho posible la mejora de la gestión de accesos en viviendas, proporcionando a los propietarios una herramienta eficiente y segura para compartir claves digitales con otros usuarios. Esto ha sido posible gracias a la implementación de tecnologías modernas y la adopción de prácticas robustas de desarrollo de software.

La fase de implementación se dividió en etapas clave, comenzando con la definición clara de requisitos y la elección estratégica de tecnologías como React para el frontend y NestJS para el backend. Esta decisión facilitó la creación de una interfaz de usuario intuitiva y responsiva, además de asegurar un backend escalable y seguro para manejar las operaciones críticas de gestión de accesos.

Los resultados obtenidos durante las pruebas y la validación del sistema fueron consistentes con las expectativas iniciales y los criterios de éxito del proyecto. La aplicación demostró ser eficaz en la gestión centralizada de múltiples cerraduras y propiedades, proporcionando a los administradores una vista consolidada y herramientas robustas para la administración de usuarios y accesos. Los huéspedes, por su parte, se beneficiaron de una experiencia simplificada y segura para encontrar y acceder a las propiedades, respaldada por un mapa interactivo y el acceso seguro a las claves digitales desde la aplicación.

En conclusión, este proyecto no solo logró cumplir con sus objetivos técnicos y funcionales, sino que también estableció una base para futuras innovaciones en la gestión de accesos.

6.2. Propuestas de Mejora

Como en todo proceso de desarrollo, existen áreas que requieren mejoras, algunas debido a limitaciones de tiempo para realizar revisiones exhaustivas y otras derivadas de la familiarización aún en proceso con la tecnología utilizada. Los aspectos a mejorar incluyen:

- Implementar mecanismos adicionales de seguridad, como autenticación multifactor (MFA) para los usuarios, o incluso la exploración de tecnologías emergentes en seguridad informática para garantizar la protección continua de datos sensibles y comunicaciones.
- Permitir una mayor personalización de las funcionalidades y configuraciones por parte de los administradores, de modo que puedan adaptar la aplicación según las necesidades específicas de cada propiedad. Esto podría incluir la capacidad de establecer horarios de acceso específicos o reglas personalizadas para cada usuario.
- Continuar refinando la interfaz de usuario y la experiencia del usuario para hacerla aún más intuitiva y fácil de usar. Realizar pruebas de usabilidad adicionales con usuarios reales para identificar y corregir cualquier punto de fricción que pueda surgir durante la interacción con la aplicación.
- Implementar herramientas analíticas que proporcionen a los usuarios insights sobre el uso de las cerraduras y las estadísticas de acceso. Esto podría ayudar a optimizar la gestión de las propiedades y mejorar la toma de decisiones basadas en datos.
- Ofrecer recursos educativos y soporte técnico continuo para usuarios, asegurando que puedan aprovechar al máximo todas las funcionalidades y capacidades de la aplicación.

Estas mejoras tienen como objetivo mejorar la funcionalidad, usabilidad y seguridad de la aplicación, consolidando aún más su papel como una herramienta valiosa en el panorama de gestión de claves.

Capítulo 7

Summary and Conclusions

7.1. Conclusions

During the development of this project, the fundamental objectives set initially were successfully achieved. The designed application has effectively enhanced access management in homes, providing property owners with an efficient and secure tool to share digital keys with other users. This was made possible through the implementation of modern technologies and robust software development practices.

The implementation phase was divided into key stages, starting with clear requirements definition and the strategic choice of technologies such as React for the frontend and NestJS for the backend. This decision facilitated the creation of an intuitive, responsive user interface and ensured a scalable, secure backend to handle critical access management operations.

The results obtained during system testing and validation were consistent with the initial expectations and project success criteria. The application proved effective in centrally managing multiple locks and properties, providing users with a consolidated view and robust tools for user and access management. Guests, on the other hand, benefited from a simplified and secure experience in finding and accessing properties, supported by an interactive map and secure access to digital keys from the application.

In conclusion, this project not only achieved its technical and functional objectives but also established a foundation for future innovations in access management.

7.2. Proposals for Improvement

As with any development process, there are areas that require improvement, some due to time constraints for thorough reviews and others

stemming from ongoing familiarity with the technology used. Areas for improvement include:

- Implementing additional security mechanisms, such as multi-factor authentication (MFA) for users, or exploring emerging technologies in cybersecurity to ensure continuous protection of sensitive data and communications.
- Allowing greater customization of functionalities and settings by administrators, so they can tailor the application to the specific needs of each property. This could include the ability to set specific access schedules or personalized rules for each user.
- Continuing to refine the user interface and user experience to make it even more intuitive and user-friendly. Conducting additional usability tests with real users to identify and address any friction points that may arise during interaction with the application.
- Implementing analytics tools that provide users insights into lock usage and access statistics. This could help optimize property management and enhance data-driven decision-making.
- Providing ongoing educational resources and technical support forusers, ensuring they can fully leverage all functionalities and capabilities of the application.

These improvements aim to enhance the application's functionality, usability, and security, further solidifying its role as a valuable tool in the key management landscape.

Bibliografía

[1] August Home, "August smart lock + connect." https://august.com/products/august-smart-lock-connect. Accessed: 2024-06-13.

[2] Yale Home, "Yale assure lock 2 plus with wi-fi and apple home keys." https://shopyalehome.com/products/yale-assure-lock-2-plus-with-wi-fi-and-apple-home-keys?variant=41587602358404. Accessed: 2024-06-13.

[3] Schlage, "Schlage smart locks with the schlage app." https://www.schlage.com/en/home/smart-locks/schlage-app.html. Accessed: 2024-06-13.

[4] Lightbee Corporation, "Lightkey by lightbee corporation." https://lightbeecorp.com/tecnologia/lightkey/. Accessed: 2024-06-13.

[5] NestJS Contributors, "Nestjs documentation." https://docs.nestjs.com/. Accessed: 2024-06-17.

[6] "GraphQL." https://graphql.org/learn/. Accessed: 2024-06-17.

[7] "TypeORM." https://typeorm.io/. Accessed: 2024-06-17.

[8] Imagina Formación, "Qué es el Virtual DOM de React." https://imaginaformacion.com/tutoriales/que-es-el-virtual-dom-de-react. Accessed: 2024-06-17.

[9] T.-H. Do and M. Yoo, "Potentialities and challenges of vlc based outdoor positioning," in *2015 International Conference on Information Networking (ICOIN)*, pp. 474–477, IEEE, 2015.

[10] J. Rufo, V. Guerra, M. Luna, J. Farmer, and D. O'Brien, "Low power control access system based on vlc for industrial applications," in *IECON Proceedings (Industrial Electronics Conference)*, vol. 2022-October, IEEE Computer Society, 2022.

[11] Netlify, "Netlify Documentation." https://docs.netlify.com/. Accessed: 2024-06-17.

[12] Render, "Render Documentation." https://docs.render.com/. Accessed: 2024-06-17.

[13] Flachmann und Heggelbacher, "Manual de Docklight." https://docklight.de/manual/index.html?docklightscripting-overview.html. Accessed: 2024-06-17.