UCAB Universidad Católica ANDRÉS BELLO

Puerta inteligente basada en el concepto del internet de las cosas

TRABAJO ESPECIAL DE GRADO

presentado ante la

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

como parte de los requerimientos para optar al título de

INGENIERO EN INFORMÁTICA

REALIZADO POR	Alonso Araujo Yormir Sayegh
PROFESOR GUÍA	Ing. Honack Villanueva
FECHA	Junio de 2015

UCAB Universidad Católica ANDRÉS BELLO

FACULTAD DE INGENIERÍA
ESCUELA DE INFORMÁTICA

Este Jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado: 20 (VEINTE)

J U R A D O E X A M I N A D O R

Firma:
Nombre: Hanack Villanueva

Firma:
Nombre: Valeria León

Firma:
Nombre: BIAGIO CAMIE

REALIZADO POR

PROFESOR GUIA

FECHA

Dedicatoria

Este trabajo está dedicado a mi familia Lucia, Ender, Anameli, Gabriela y Alberico quienes me han impulsado para alcanzar mis metas; a mi novia Samantha quien ha llenado de alegría mis días desde hace ya tanto tiempo; a mi compañero y amigo Samir con quien he tenido la dicha de trabajar en proyectos fascinantes; a mis buenos amigos Giuseppe, Eduardo, Leonardo, Liliana, Jonathan, Jorge, Karem, Picho quienes me han acompañado durante todo este camino; y a los profesores Ricardo, Susana, Lúcia, Honack, Gerardo, David, Genesio, quienes me han formado e inspirado a aprender cada día más.

A todos, muchas gracias.

– Alonso Araujo

Dedico este trabajo a Iris, Samir, Antonieta, Camila y Yeritza que día a día me ayudan a alcanzar mis metas; mi familia y las personas que no son familia pero forman parte de mi vida como si lo fueran; a Karem, mi novia, que ha estado conmigo cuando más la necesito, a Alonso un gran compañero con el que trabajar en equipo es destacable; a mis amigos Jorge, Paul, Génesis, Liliana, Gary, Leonardo, Oswaldo y Miguelangel, quienes siempre han estado allí cuando se les necesita y cuando no.

Agradecido con todos ustedes.

– Yormir Sayegh

Agradecimientos

Queremos agradecer a todas las personas que apoyaron a la realización de este trabajo de grado especialmente a Honack Villanueva por ofrecernos su guía como tutor del proyecto, Susana García por ofrecer su apoyo en todo momento, Lúcia Cardoso y Adrianna Adrian por guiarnos hacia el correcto diseño de la base de datos, Biagio Cante por agregar valor a nuestra solución con sus propuestas, Thomas Maragno por compartir sus conocimientos sobre el internet de las cosas, Fabian Correa por alentarnos y motivarnos al construir nuestro proyecto, Antonio Farchakh por facilitarnos los medios para trasladar la puerta, Samir Sayegh por ayudarnos con la logística del traslado y construcción del prototipo, Liliana Herrera por alentarnos y corregir errores de redacción, Samantha Andrades y Karem Segura gracias por estar a nuestro lado durante todo este tiempo apoyándonos para culminar el proyecto con éxito.

A todos ustedes agradecemos su apoyo y contribución directa en la realización de este trabajo de grado.

– Alonso Araujo y Yormir Sayegh

Sinopsis

El internet de las cosas es una de las tendencias tecnológicas con mayor potencial en la actualidad. Esta ha impulsado una gran cantidad de iniciativas tecnológicas alrededor del mundo, que buscan cambiar por completo la forma en cómo interactuamos con dispositivos tan tradicionales como una cafetera, una nevera, un televisor o incluso un bombillo; apalancándose en el internet para crear una red de comunicación entre dispositivos que de manera autónoma interactúan entre sí.

Se espera que para 2018 más de la mitad del tráfico en la red provenga directamente de dispositivos del internet de las cosas, y aunque esta tendencia luce prometedora a largo plazo, en la actualidad existen una gran cantidad de retos técnicos asociados a la creación de nuevos dispositivos diseñados tomando en cuenta este concepto.

En este trabajo de grado se describe el proceso de diseño y construcción de un prototipo de puerta inteligente basada en el concepto del internet de las cosas, como también se presenta la arquitectura técnica que soporta la solución.

Esta puerta inteligente aprovecha sus componentes tecnológicos para brindar capacidades aumentadas de seguridad y facilidad de uso, al mismo tiempo que hace uso del internet para comunicarse e interactuar con otros dispositivos demostrando de manera práctica las posibilidades habilitadas por la llamada internet de las cosas.

Tabla de contenidos

Capítulo I – Introducción **1**
1.1 Declaración de la oportunidad 1
1.2 Objetivo general 2
1.3 Objetivos específicos 2
1.4 Alcance 3
1.4.1 Prototipo Puerta 3
1.4.2 Servidor central 4
1.4.3 Aplicación móvil 5
1.4.4 Puerta inteligente para el internet de las cosas 6
1.4.5 Investigación y evaluaciones 7
1.4.6 Prueba de concepto 7
1.5 Limitaciones 8
1.6 Justificación 9
Capítulo II – Marco referencial **11**
2.1 ¿Qué es el internet de las cosas? 11
2.2 Evolución del término 11
2.3 Retos técnicos asociados al internet de las cosas 13
Capítulo III – Metodología **16**
3.1 Feature Driven Development 16
3.2 Mejores prácticas en FDD 19
3.3 Beneficios de FDD 21
Capítulo IV – Diseño y desarrollo **22**
4.1 Iteración cero (0) - Fase inicial de diseño y planificación 22
4.2 Iteración uno (1) - Construcción de prototipo de la puerta 24
4.2.1 Diseño 24
4.2.2 Implementación 25
4.2.3 Entregable 27
4.3 Iteración dos (2) - Desarrollo de software de la puerta 28
4.3.1 Diseño 28

4.3.1.1 Evaluación de tecnologías de comunicación 28
4.3.2 Implementación 29
4.3.2.1 Evaluación de métodos de autenticación 32
4.3.3 Entregable 34
4.4 Iteración tres (3) - Desarrollo de servidor central 36
4.4.1 Diseño 36
4.4.2 Implementación 37
4.4.3 Entregable 39
4.5 Iteración cuatro (4) - Desarrollo de aplicación móvil 41
4.5.1 Diseño 41
4.5.2 Implementación 43
4.5.3 Entregable 44
4.6 Iteración cinco (5) - Desarrollo de dispositivos externos 45
4.6.1 Diseño 45
4.6.2 Implementación 46
4.6.3 Entregable 47
4.7 Pruebas de concepto 48
Capítulo V – Arquitectura de la solución 50
5.1 Arquitectura de solución para el internet de las cosas 50
5.1.1 Interoperabilidad 51
5.1.2 Escalabilidad 53
5.1.3 Autonomía del dispositivo 56
5.1.4 Seguridad y privacidad 57
Capítulo VI – Resultados 59
6.1 Puerta inteligente para el internet de las cosas 59
6.2 Prototipo de puerta 60
6.3 Aplicación móvil 62
6.4 Servidor central 63
6.6 Interacción con dispositivos del internet de las cosas 64
6.7 Cobertura de objetivos planteados 65
Capítulo VII – Conclusiones 68

7.1 Conclusiones 69
7.2 Recomendaciones 73
Referencias 75

Índice de figuras

Figura 1: Modelo general de solución 6
Figura 2: Flujo de actividades FDD 17
Figura 3: Iteración cero en FDD 22
Figura 4: Roadmap del proyecto 24
Figura 5: Diagrama de zonas 25
Figura 6: Construcción del prototipo 26
Figura 7: Circuito apertura de pestillo 26
Figura 8: Modelos faciales generados con Fisherfaces 30
Figura 9: Diagrama de componentes para servidor central 39
Figura 10: Maquetas de interfaz para aplicación móvil 42
Figura 11: Diagrama de secuencia para proceso de registro 43
Figura 12: Diagrama de despliegue de la solución 50
Figura 13: Diagrama de clases para servidor central 52
Figura 14: Arquitectura de la solución con dispositivos externos 53
Figura 15: Diagrama de componentes para servidor central 55
Figura 16: Diagrama de secuencia para apertura remota 58
Figura 17: Modelo general del producto resultante 59
Figura 18: Prototipo de puerta 61
Figura 19: Pantalla de inicio 62
Figura 20: Métodos de comunicación de la solución 64

Índice de tablas

Tabla 1: Crecimiento de dispositivos conectados 13
Tabla 2: Funcionalidades completadas para iteración 1 27
Tabla 3: Control de avance por área funcional para iteración 1 27
Tabla 4: Evaluación de tecnologías de comunicación 29
Tabla 5: Evaluación de métodos de autenticación según velocidad. 32
Tabla 6: Evaluación de métodos de autenticación en base a seguridad 34
Tabla 7: Funcionalidades completadas para iteración 2 35
Tabla 8: Control de avance por área funcional para iteración 2 35
Tabla 9: Funcionalidades completadas para iteración 3 40
Tabla 10: Control de avance por área funcional para iteración 3 40
Tabla 11: Funcionalidades completadas para iteración 4 44
Tabla 12: Control de avance por área funcional para iteración 4 44
Tabla 13: Funcionalidades completadas para iteración 5 47
Tabla 14: Control de avance por área funcional para iteración 5 47

"Las tecnologías más impresionantes son las que no se perciben. Se esconden a sí mismas entre el tejido de la vida cotidiana hasta que son simplemente indistinguibles."

– Mark Weiser
The Computer for the 21st Century

Capítulo I - Introducción

1.1 Declaración de la oportunidad

En la actualidad una de las tendencias tecnológicas que ha cobrado mayor fuerza a nivel mundial es la llamada "internet de las cosas" [1]. Ésta describe un futuro en el que todo dispositivo de uso cotidiano dispondrá de capacidades de procesamiento y conexión a internet, que le servirán no solo para compartir información detallada con su dueño, sino también para comunicarse con otros dispositivos de manera autónoma creando de esta manera la red que le da nombre a esta tendencia [2].

El internet de las cosas está siendo construido en estos momentos a nivel mundial y cada vez más ingenieros y desarrolladores están creando nuevos dispositivos que posiblemente formarán parte del día a día en un futuro cercano. Con desarrollos que van desde bombillos inteligentes que permiten el encendido remoto y consumo eficiente de la energía eléctrica [3], termostatos conectados que usan información de pronósticos del tiempo para calcular la mejor temperatura dentro del hogar [4], hasta papeleras públicas que se comunican directamente con la alcaldía para pedir ser vaciadas [5]. Se pronostica que el internet de las cosas sea uno de los campos de la informática con mayor crecimiento y adopción en los próximos años [6].

Existe un largo camino para lograr lo descrito por esta tendencia, ya que deben diseñarse y construirse una gran cantidad de dispositivos inteligentes que tomen en cuenta y superen los retos técnicos asociados a implementar

soluciones para el internet de las cosas [7]. Es en base a esta premisa que se concibe este trabajo de grado, en el que se busca implementar el concepto del internet de las cosas en un objeto tan tradicional como lo es una puerta.

1.2 Objetivo general

Diseñar, desarrollar y construir un prototipo de puerta inteligente que brinde seguridad a sus usuarios, mediante el apoyo de la tecnología; y que pueda comunicarse e interactuar con otros dispositivos vía internet.

1.3 Objetivos específicos

- Construir una puerta de seguridad, integrando electrónicamente los siguientes dispositivos: nodo de cómputo, cámara, lector de huellas dactilares, pestillo eléctrico y teclado numérico.
- Desarrollar una aplicación para el nodo de cómputo que controle la apertura del pestillo, utilizando tecnologías de autenticación como reconocimiento facial, lectura de huellas dactilares y contraseña numérica.
- Desarrollar una aplicación de servidor central que permita gestionar perfiles de usuario, preferencias, y la comunicación e interacción entre la puerta y otros dispositivos vía internet.
- Desarrollar una aplicación móvil que permita la configuración y apertura remota de la puerta.
- Evaluar servicios de mensajería en la capa de software entre la puerta y el servidor central en base a seguridad e implementar el mejor de ellos.

- Evaluar combinaciones de métodos de seguridad implementados (clave, reconocimiento facial y huellas dactilares) en base a seguridad y velocidad.
- Construir un prototipo de la puerta a escala real.
- Crear una prueba de concepto del proyecto en donde se demuestren dos escenarios de funcionalidad.

1.4 Alcance

Este trabajo de grado plantea la creación de una puerta que integre distintos componentes electrónicos, que permitan añadir capacidades de seguridad avanzadas apoyadas en tecnología; además, la misma podrá conectarse a una red WiFi para comunicarse con un servidor central que facilitará la comunicación entre la puerta y otros dispositivos vía internet. De esta manera la puerta no solo será capaz de combinar distintos métodos de autenticación, sino también de interactuar con otros dispositivos vía internet habilitando nuevos casos de uso en base a posibles interacciones con dichos dispositivos.

1.4.1 Prototipo Puerta

Esta sección describe en detalle el alcance de los siguientes objetivos:

- Construir una puerta de seguridad, integrando electrónicamente los siguientes dispositivos: nodo de cómputo, cámara, lector de huellas dactilares, pestillo eléctrico y teclado numérico.

- Desarrollar una aplicación para el nodo de cómputo que controle la apertura del pestillo, utilizando tecnologías de autenticación como reconocimiento facial, lectura de huellas dactilares y contraseña numérica.
- Construir un prototipo de la puerta a escala real.

La puerta contará con componentes tecnológicos que le permitan brindar capacidades aumentadas de seguridad y comunicarse con otros dispositivos a través de internet. De esta manera la puerta será capaz de autenticar al usuario con tecnologías como reconocimiento facial, lectura de huellas dactilares y contraseñas numéricas, aumentando la seguridad en la misma.

Una característica importante de la arquitectura de la solución es que el nodo de cómputo interno debe almacenar toda información sensible (claves, huellas, y fotos) y realizar todo el procesamiento necesario para la apertura de la puerta, por lo que las capacidades de autenticación y apertura estarán disponibles sin conexión a internet.

La puerta contará con la capacidad de identificar si se encuentra abierta o cerrada utilizando un sensor magnético.

1.4.2 Servidor central

Esta sección describe en detalle el alcance de los siguientes objetivos:

- Desarrollar una aplicación de servidor central que permita gestionar perfiles de usuario, preferencias, y la comunicación e interacción entre la puerta y otros dispositivos vía internet.

Estará encargado de gestionar los perfiles de usuario y sus configuraciones personales, y de manejar las conexiones e interacciones de la puerta con otros dispositivos. Se espera que este servidor pueda migrarse fácilmente a un modelo en la nube para aprovechar sus beneficios.

Este servidor brindará distintos servicios a través de un API; consumirá servicios de otros dispositivos, y se comunicará de manera segura con la puerta y la aplicación móvil. A través del API distintos dispositivos podrán interactuar con la puerta.

1.4.3 Aplicación móvil

Esta sección describe en detalle el alcance de los siguientes objetivos:

- Desarrollar una aplicación móvil que permita la configuración y apertura remota de la puerta.

La aplicación móvil estará encargada de la configuración y personalización de los servicios que brindará la puerta. Funcionará como panel de control de la misma y permitirá realizar funcionalidades extendidas tales como:

- Visualización remota de la cámara integrada en la puerta.
- Apertura remota de la puerta.
- Activación de botón de pánico.
- Consulta de histórico de entradas detallado.

1.4.4 Puerta inteligente para el internet de las cosas

La suma de los componentes anteriormente descritos resulta en la solución completa de puerta inteligente para el internet de las cosas, en donde la puerta contará con un nodo de cómputo autónomo que con una conexión a internet podrá comunicarse con distintos objetos como celulares, sensores y dispositivos inteligentes, enlazada por medio del servidor central y controlada remotamente por la aplicación móvil.

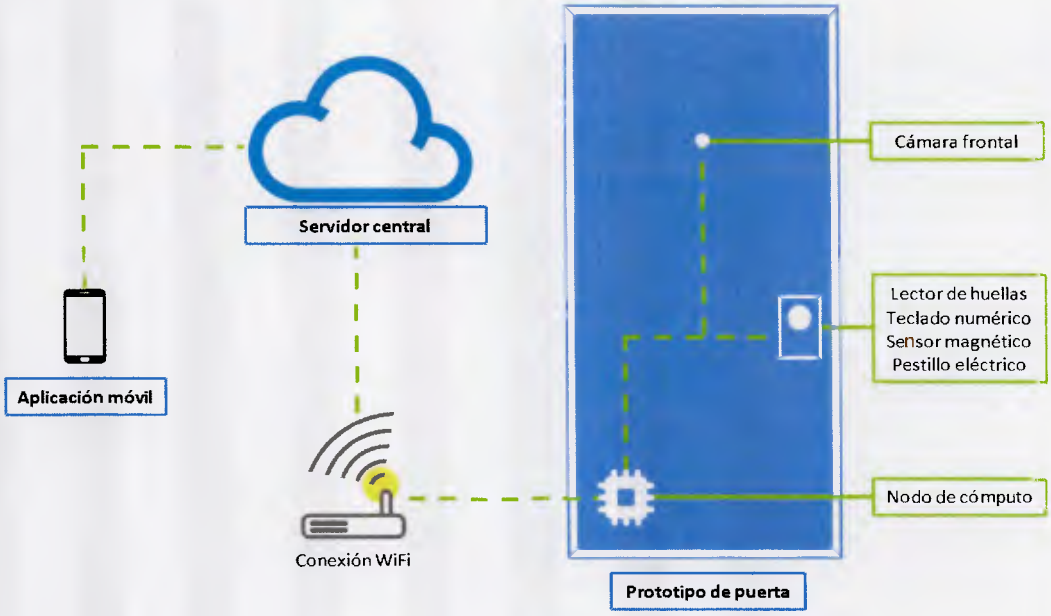


Figura 1: Modelo general de solucion
Fuente: Elaboración propia

1.4.5 Investigación y evaluaciones

Esta sección describe en detalle el alcance de los siguientes objetivos:

- Evaluar servicios de mensajería en la capa de software entre la puerta y el servidor central en base a seguridad e implementar el mejor de ellos.

Durante la ejecución del proyecto también se realizarán una serie de evaluaciones sobre las tecnologías a utilizar, siendo sus objetivos:

1. Evaluación de tecnologías de comunicación entre puerta y servidor central en base a seguridad de la conexión.
2. Evaluación de combinaciones de métodos de autenticación implementados en base a velocidad y seguridad del método.
3. Investigación y evaluación de hardware para el sistema de apertura de la puerta en base a precio, seguridad y facilidad de implementación.

1.4.6 Prueba de concepto

Esta sección describe en detalle el alcance de los siguientes objetivos:

- Crear una prueba de concepto del proyecto en donde se demuestren dos escenarios de funcionalidad.

Al finalizar el proyecto se realizará una prueba de concepto para demostrar de forma práctica el producto resultante del trabajo de grado. En esta prueba de concepto se pretende mostrar el funcionamiento de la puerta con dos escenarios:

- **Reproductor de música**: un usuario es autenticado por la puerta utilizando reconocimiento facial, ésta lo reconoce y le permite el acceso; lo que activa la reproducción de una canción predeterminada en un reproductor de música de una computadora externa.
- **Correo electrónico automático**: un usuario es autenticado por la puerta utilizando reconocimiento facial, lectura de huellas dactilares y contraseña numérica. Luego de permitirle el acceso envía de manera automática un correo electrónico a un destinatario predeterminado.

Para esta prueba de concepto se construirá un modelo de la puerta a escala real, que contará con todas las funcionalidades descritas en este apartado. En ambos escenarios los usuarios serán diferentes para demostrar la capacidad de la puerta de configurar interacciones según el usuario que entra.

1.5 Limitaciones

- La cantidad de personas registradas en la puerta es finita y está determinada por la capacidad de almacenamiento del nodo de cómputo presente en la puerta.
- Si la conexión a internet del nodo de cómputo en la puerta falla, las características de funcionamiento que la requieran no podrán ser utilizadas.
- La puerta necesita estar conectada a una entrada de corriente eléctrica y en caso de perder la fuente de poder por un tiempo prolongado y se agote la batería interna, la puerta no podrá ser abierta con el sistema eléctrico.

- De presentar una falla en el nodo de cómputo presente en la puerta, el sistema de apertura eléctrico no podrá ser utilizado.
- De no haber ninguna fuente de luz al frente de la cámara, no se podrá realizar el reconocimiento facial para la apertura de la puerta.
- El nodo de cómputo debe tener una red WiFi preconfigurada para permitir la primera conexión con el dispositivo móvil.

1.6 Justificación

Si bien el concepto del internet de las cosas propone un futuro con grandes beneficios y facilidades para los usuarios de dichos dispositivos inteligentes conectados, existen dificultades técnicas que en la actualidad no han sido resueltas y que presentan un reto para la comunidad tecnológica mundial. Es por esta razón que en la actualidad una gran cantidad de empresas e individuos están trabajando para definir estándares que faciliten el desarrollo de este tipo de dispositivos.

Para volver una realidad lo descrito por el internet de las cosas, también es necesario crear una gran cantidad de dispositivos que apliquen el concepto del internet de las cosas. Las características y requerimientos técnicos de estos dispositivos difieren de los sistemas y dispositivos tradicionales, y es por esta razón que muchos de los paradigmas y arquitecturas clásicas de sistemas de información están presentando dificultades para soportar este tipo de soluciones.

Son estas necesidades las que motivan al equipo a llevar a cabo este trabajo de grado, en donde no solo se pretende diseñar un objeto inteligente tomando en cuenta lo descrito en el concepto del internet de las cosas, sino que al mismo tiempo se busca presentar un diseño de arquitectura robusto que permita superar gran parte de los retos principales asociados a la tendencia.

Capítulo II - Marco referencial

2.1 ¿Qué es el internet de las cosas?

El internet de las cosas es un término utilizado para describir las posibilidades habilitadas por el uso en conjunto del internet y el creciente número de dispositivos conectados a nivel mundial.

Siendo el internet de las cosas un concepto en plena maduración, puede destacarse la definición con mayor adopción a nivel global, de Kevin Ashton, que dicta una visión más abierta y general sobre el concepto del internet de las cosas. Ashton expresa que el internet de las cosas se define como la red creada por la interacción e intercambio de datos entre dispositivos de manera autónoma vía internet [8].

Estos dispositivos no solo se mantienen conectados a la red para ser controlados remotamente, sino que aprovechan esta conectividad para extender sus funcionalidades apoyándose en la interacción con otros dispositivos conectados de manera autónoma, creando una gran cantidad de nuevos usos y servicios.

2.2 Evolución del término

El término "internet de las cosas" fue utilizado por primera vez en 1.999 por Kevin Ashton, director de Auto-ID Center del Massachusetts Institute of Technology (MIT), en una presentación comercial para P&G sobre uso de

etiquetas RFID en la cadena de producción para registrar y controlar sus productos [8].

Ese mismo año Neil Gershenfeld, profesor del MIT, publica un libro titulado *When Things Start to Think* (Cuando las cosas comienzan a pensar) que describe cómo cualquier dispositivo puede apoyarse en el internet para crear un sistema con otros dispositivos [9].

Durante el período 2000-2005 distintos medios especializados comienzan a popularizar el término incluyéndolo en sus reportes [10], lo cual despierta el interés en compañías que comienzan a comercializar productos asociándolos al internet de las cosas. Es a finales del 2005 cuando es utilizado el término de manera formal con un detallado reporte publicado por la Unión Internacional de Telecomunicaciones de las Naciones Unidas [11].

En 2008 el Consejo de Inteligencia Nacional de Estados Unidos, cataloga el internet de las cosas como una "tecnología civil disruptiva" [12] posicionándolo como aspecto clave para los intereses de la nación hasta 2025.

El Grupo de Soluciones de Negocio de Internet de Cisco nombra el 2009 como el año de nacimiento para el internet de las cosas [13], apoyado en que en ese momento existen significativamente más dispositivos conectados a internet que la población global total, sus cifras indican que para 2010 existían 12,5 mil millones de dispositivos conectados versus 6,8 mil millones de personas a nivel mundial.

En la actualidad es considerada una de las tendencias tecnológicas más importantes del momento y Cisco proyecta que la cantidad de dispositivos conectados a internet superará los 50 mil millones para 2020 [13].

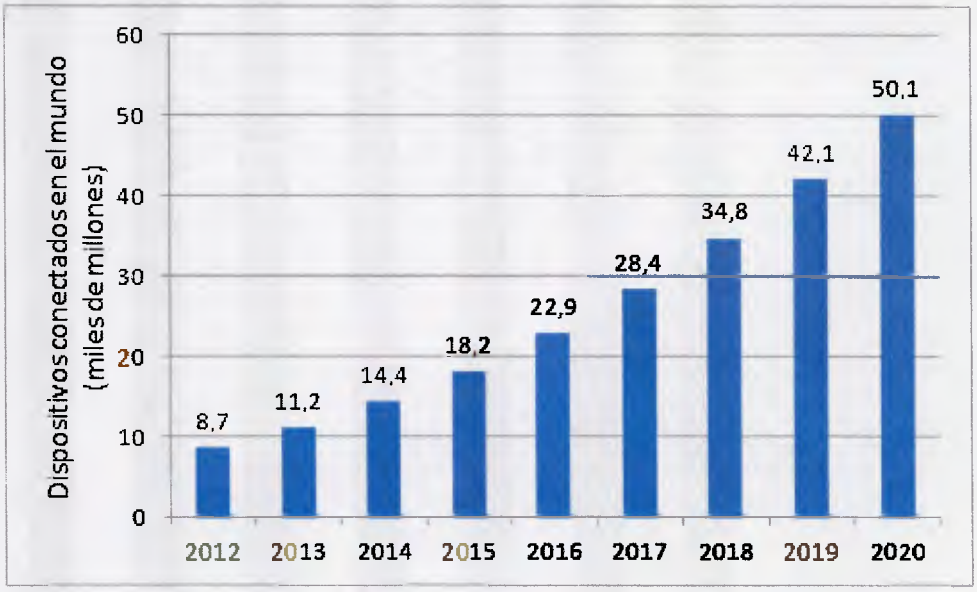


Tabla 1: Crecimiento de dispositivos conectados
Fuente: Adaptación Cisco 2007

2.3 Retos técnicos asociados al internet de las cosas

Si bien las posibilidades que propone el internet de las cosas son amplias, también lo son los retos técnicos que trae consigo el implementar este concepto utilizando las tecnologías existentes en la actualidad. Frente a esta visión, se han realizado múltiples esfuerzos de investigación e ingeniería como el de Khan, R. et al. (2012) con el trabajo titulado *Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges* en donde se describe el estado del internet de las cosas y los retos técnicos asociados, como también posibles soluciones [7]. También resalta el trabajo de investigación titulado *Comparative Analysis of IoT Architectures*

publicado por Guthikonda, R. et al. (2014) en donde se realiza una comparación detallada de arquitecturas técnicas existentes aplicadas al internet de las cosas [14].

Al estudiar este tipo de trabajos queda claro que en la actualidad aún no existe un estándar o mejor práctica definida y aceptada para diseñar arquitecturas que den soporte a este tipo de soluciones, que por su naturaleza distribuida presentan grandes retos técnicos al requerir de interoperabilidad, escalabilidad, autonomía en sus dispositivos, y seguridad; a niveles o en situaciones diferentes a las de una plataforma tradicional. A continuación se describen en detalle cada uno de los retos técnicos:

- **Interoperabilidad:** en una arquitectura para el internet de las cosas, existen distintos actores que comprenden usuarios y objetos. Un actor puede desempeñar múltiples roles dentro de la arquitectura, tales como, monitoreo de los recursos disponibles, proveedor de datos, consumidor de datos, proveedor de servicios, orquestador de interacciones, interfaz con usuario final, entre otros. Lograr una integración transparente entre los diferentes actores es crucial para contemplar la visión del internet de las cosas. La integración entre los diferentes objetos puede ser compleja, sobre todo cuando cada actor se maneja e implementa de manera diferente.

- **Escalabilidad:** el rápido crecimiento de las tecnologías embebidas está dando lugar a un enorme despliegue de dispositivos inteligentes con sensores, actuadores, entre otros. A medida que el número de

dispositivos aumenta, los datos producidos por estos dispositivos tienden a crecer de manera exponencial. Por lo tanto, el procesamiento de altos volúmenes de datos y dispositivos es uno de los principales retos en una arquitectura del internet de las cosas.

- **Autonomía del dispositivo:** si bien una solución del internet de las cosas aprovecha la comunicación entre múltiples dispositivos y no debe esperarse obtener el mismo resultado de la solución sin la conectividad entre los mismos, se pueden presentar escenarios en donde se requiera de cierto grado de autonomía en el dispositivo al verse limitado o sin conexión a la red. Esto presenta un reto importante a la hora de diseñar dispositivos que cuenten con capacidades funcionales sin necesidad de estar conectados a internet.

- **Seguridad y privacidad:** Debido al gran número de actores involucrados en este tipo de arquitecturas, garantizar la autenticación de usuarios y dispositivos, y el control de acceso a datos, debe ser una característica fundamental para cada uno de los actores. Para desarrollar una arquitectura robusta del internet de las cosas, la seguridad de la información, privacidad y protección de datos debe ser abordada con prioridad y detalle.

Capítulo III - Metodología

La metodología seleccionada para la elaboración de este trabajo de grado fue *Feature Driven Development* (FDD). FDD es una metodología ágil que busca agrupar las mejores prácticas utilizadas en la industria del desarrollo de software y aprovechar las características presentes tanto en enfoques ágiles como tradicionales [15]. Su fortaleza está en el desarrollo de proyectos de tecnología en donde los requerimientos ya son claros y están bien definidos. Dada la naturaleza de este proyecto, en donde los requerimientos fueron definidos por el mismo equipo de desarrollo, se decidió utilizar FDD en busca de cubrir el ciclo de vida del proyecto de manera ágil. A continuación se describe en detalle la metodología y las prácticas aplicadas en la realización de este proyecto.

3.1 Feature Driven Development

Feature Driven Development (FDD) es una metodología ágil iterativa e incremental, que forma parte de la Agile Alliance y fue creada por Jeff De Luca en 1997 [16]. Esta metodología se enfoca en la entrega ágil de características funcionales del proyecto de manera iterativa en ciclos cortos de tiempo.

Es una metodología guiada por modelos y basada en procesos iterativos que se componen por cinco actividades básicas. Para mantener una gestión precisa de los avances del proyecto, la metodología demanda la definición de hitos y entregables que demuestren el progreso realizado en cada iteración.

Las primeras tres actividades de la metodología constituyen la fase inicial del proyecto, mientras que las últimas dos son aplicadas para cada una de las iteraciones definidas, como puede apreciarse en la figura a continuación.

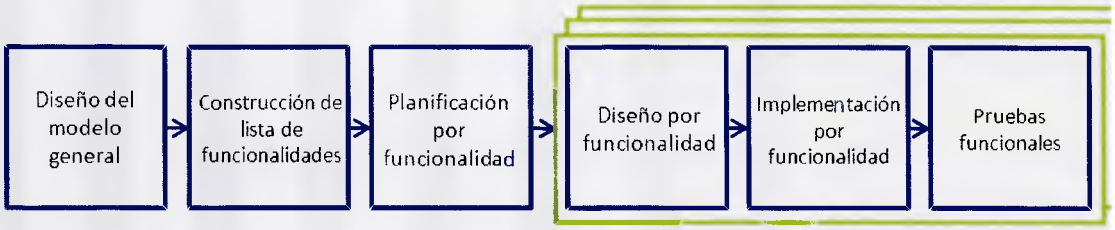


Figura 2: Flujo de actividades FDD
Fuente: Adaptación de Jeff de Luca

A continuación se explican en detalle cada una de las actividades:

- **Diseño del modelo general:** un proyecto FDD empieza con un diseño de alto nivel para el alcance y contexto del mismo. Luego, se divide este alcance en áreas y se diseña un modelo de dominio para cada una de estas áreas. Los modelos de dominio comienzan a un alto nivel y son detallados progresivamente hasta llegar a su versión final.

- **Construcción de lista de funcionalidades:** el conocimiento obtenido durante la fase de modelado inicial es utilizado para identificar una lista de funcionalidades, al mismo tiempo se agrupan dichas funcionalidades en actividades para cada una de las áreas definidas. Las funcionalidades son pequeñas funciones valoradas por el cliente, expresadas en la forma de "<acción><resultado><objeto>". Las funcionalidades no deben tomar más de dos semanas en ser desarrolladas, de lo contrario deberán ser divididas en funcionalidades de menor alcance.

- **Planificación por funcionalidad:** luego de completada la lista de funcionalidades, el próximo paso es la creación de un plan de desarrollo; este plan es dividido en iteraciones y deben asignarse programadores responsables para cada una de las funcionalidades.

- **Diseño por funcionalidad:** un paquete de diseño es creado para cada funcionalidad. El responsable del equipo de desarrollo selecciona un conjunto de funcionalidades que serán trabajadas durante las próximas dos semanas.

  En conjunto y con todo el equipo involucrado en el proceso de diseño, el responsable del equipo diseña diagramas de secuencia detallados para cada funcionalidad y detalla el modelo de dominio general. Luego de detalladas las clases y métodos asociados, se lleva a cabo una inspección de diseño.

- **Implementación por funcionalidad:** luego de realizada la inspección de diseño, se planean las actividades para el desarrollo de las funcionalidades seleccionadas. Los responsables de cada funcionalidad comienzan la implementación del código. Luego de realizadas las pruebas y una inspección de código, la funcionalidad completada es promovida a la instancia principal.

- **Pruebas funcionales:** una vez finalizada la iteración se realizan las pruebas correspondientes para garantizar la correcta implementación de las funcionalidades logradas.

3.2 Mejores prácticas en FDD

Feature Driven Development está basado en un conjunto de mejores prácticas reconocidas en la industria derivadas de la ingeniería de software, las cuales fueron adoptadas pensando en brindar el mayor beneficio al cliente a través de la entrega de funcionalidades de manera ágil. Esta combinación de prácticas y técnicas convierte a FDD en una de las metodologías ágiles más completas. A continuación se describen brevemente las mejores prácticas mencionadas:

- **Modelado de dominio orientado a objetos**: consiste en explorar y explicar el dominio del problema a resolver. El modelo de dominio resultante proporciona un marco general en el que añadir características y funciones detalladas.
- **Desarrollo de funcionalidades granulares**: cualquier funcionalidad que sea demasiado compleja para ser implementada dentro de dos semanas se descompone en funcionalidades más pequeñas hasta que cada sub-función sea lo suficientemente pequeña para ser considerada una funcionalidad válida. Esto hace que sea más fácil entregar funcionalidades correctas y ampliar o modificar el sistema de manera ágil.
- **Clase con propiedad individual**: la propiedad de clase individual significa que piezas distintas o una agrupación de código particular son asignadas a un solo dueño. El propietario es responsable de la consistencia, el rendimiento, y la integridad conceptual de la clase.

- **Equipos de funcionalidad**: un equipo de funcionalidad es un equipo pequeño formado dinámicamente que desarrolla una actividad específica. Al hacerlo, múltiples puntos de vista siempre se aplican a cada decisión de diseño, y también múltiples opciones de diseño son siempre evaluadas antes de ser elegidas.
- **Inspecciones**: las inspecciones se llevan a cabo para asegurar buena calidad del diseño y el código, principalmente mediante la detección de defectos.
- **Gestión de la configuración**: esto ayuda a identificar en el código fuente todas las funcionalidades que se han completado hasta la fecha, y al mismo tiempo a mantener un historial de los cambios en las clases a medida que los equipos de funcionalidad implementan y mejoran las mismas.
- **Compilaciones regulares**: garantizan que siempre exista una versión ejecutable y funcional del proyecto que pueda ser demostrada al cliente, y ayuda evidenciando errores de integración de código fuente de las funcionalidades en una etapa temprana.
- **Visibilidad de los avances y resultados**: reportando de manera progresiva, frecuente, apropiada y precisa los informes de avance a todos los niveles dentro y fuera del proyecto, basado en funcionalidades listas, los gerentes tienen la visibilidad necesaria para llevar en la dirección correcta al proyecto.

3.3 Beneficios de FDD

Feature Driven Development es una metodología de alto nivel que sobresale en proyectos en donde los requerimientos funcionales están bien definidos y que cuentan con una arquitectura modular. El enfoque de la metodología se centra en los aspectos técnicos y de implementación, basándose en mejores prácticas de desarrollo de software y en utilizar la menor cantidad de documentos necesarios para lograr la entrega ágil de características funcionales de la solución de manera progresiva.

En el presente trabajo de grado los requerimientos funcionales están bien definidos y son propios del equipo implementador. Por otro lado, las características de arquitectura de una solución del internet de las cosas son por definición modulares. En base a estas premisas, la aplicación de FDD se presta ideal para el desarrollo de este trabajo de grado, brindando un marco metodológico claro, ágil y medible para gestionar el ciclo de vida del proyecto.

Luego de definidos los objetivos y el alcance del trabajo de grado se realizó la fase inicial de planificación siguiendo los lineamientos de la metodología seleccionada. En este apartado se detalla el ciclo de vida del proyecto describiendo el trabajo realizado desde la iteración cero (0) o fase inicial de diseño y planificación, hasta la ejecución de la prueba de concepto de la solución resultante.

4.1 Iteración cero (0) - Fase inicial de diseño y planificación

Siguiendo los lineamientos de FDD, se inició el proyecto llevando a cabo las primeras tres actividades de la metodología, también consideradas como la "iteración cero (0)", en donde se busca definir un modelo general de dominio, la lista de funcionalidades de la solución, y un plan de trabajo en base a las funcionalidades.

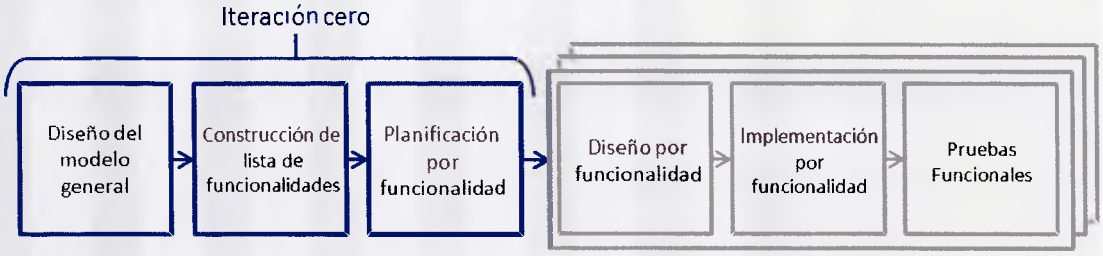


Figura 3. Iteración cero en FDD - Fuente: Adaptación de Jeff de Luca

La primera actividad realizada consistió en la creación de un modelo general de dominio. Dicho modelo se construye en base a los requerimientos del proyecto y el mismo debe cubrir el contexto y alcance total. Se construyó un modelo de dominio para cada uno de los componentes lógicos de la

arquitectura, como lo son la puerta, el servidor central, y el dispositivo móvil. Dichos modelos de dominio de alto nivel son utilizados y detallados en cada una de las iteraciones específicas durante la etapa iterativa. Estos modelos de dominio generales se pueden detallar en el Anexo B.

Una vez diseñado el modelo general de dominio, se comenzó con la construcción de la lista de funcionalidades de la solución. Dicha lista de funcionalidades es dividida jerárquicamente en áreas, actividades y funcionalidades. Se definieron tres áreas: puerta, aplicación móvil, y servidor central; para cada una de las áreas se definieron actividades generales y funcionalidades por actividad. En esta lista se encuentran definidas todas las funcionalidades que se esperan implementar en la solución, la misma puede detallarse en el Anexo A.

Por último, se comenzó con la etapa de planificación por funcionalidad. En esta etapa se construyó un plan de trabajo iterativo en donde se seleccionó un grupo de funcionalidades para cada iteración, siempre apuntando a construir las mismas de manera incremental. Para cada una de las iteraciones se asignó un responsable por funcionalidad, y de la misma manera para cada clase del modelo de dominio, siguiendo las mejores prácticas definidas por la metodología aplicada. El plan de trabajo está conformado por cinco iteraciones, que pueden detallarse en la figura a continuación.

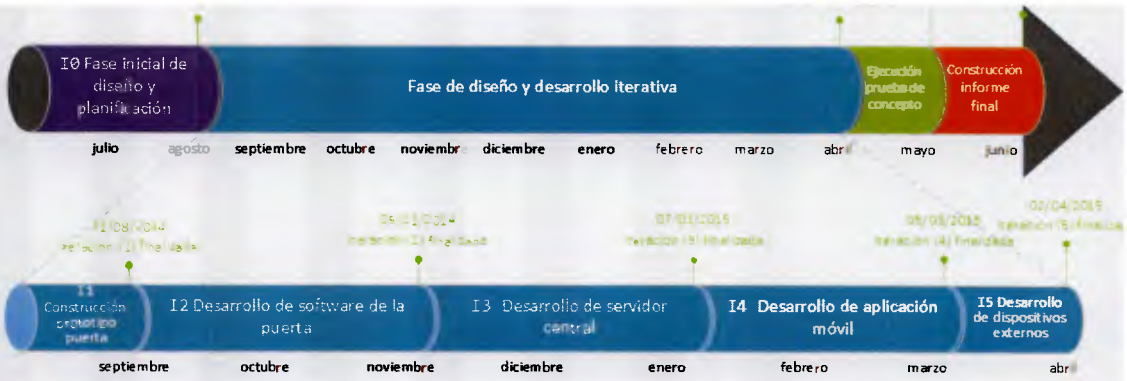


Figura 4: Roadmap del proyecto. Fuente: Elaboración propia

Luego de completada la iteración cero (0), se realizó una fase de diseño detallado, implementación y pruebas, para cada una de las iteraciones definidas en el plan de desarrollo por funcionalidad.

4.2 Iteración uno (1) - Construcción de prototipo de la puerta

La iteración uno (1) contempló todo lo referente a la construcción física del prototipo de la puerta; diseñar, construir y ensamblar los componentes de hardware necesarios que habilitan las funcionalidades definidas.

4.2.1 Diseño

Tomando en cuenta las características funcionales definidas, se listaron los componentes necesarios para el funcionamiento de la puerta, siendo estos: nodo de cómputo, cámara, detector de huellas dactilares, teclado numérico, corneta, pestillo eléctrico, circuito de apertura del pestillo eléctrico, sensor de efecto Hall, elemento para iluminación, tres transformadores de corriente, lámina protectora, y puerta de madera con su respectivo marco para su instalación.

Una vez listados los componentes necesarios, se definió su distribución en la puerta con el fin de posicionar los componentes delicados en lugares seguros con bajo impacto o movimiento. Se definieron tres zonas de despliegue dentro de la puerta:

- **Zona 1:** representa el área de visión. Esta zona debe estar a una altura óptima para efectuar el proceso de autenticación por reconocimiento facial, por esta razón se utilizó como referencia la altura promedio de la población en Venezuela [17].
- **Zona 2:** contempla los componentes de interacción del usuario y seguridad de la puerta. Se definió que esta zona estaría cercana a la manilla de la puerta ya que permite que el usuario tenga mayor comodidad para usar el detector de huellas dactilares y el teclado numérico.
- **Zona 3:** es la zona de procesamiento y componentes eléctricos. Esta zona se ubica cerca de las bisagras y en una altura media para minimizar el impacto que ocasiona el movimiento de la puerta al abrir y cerrar.

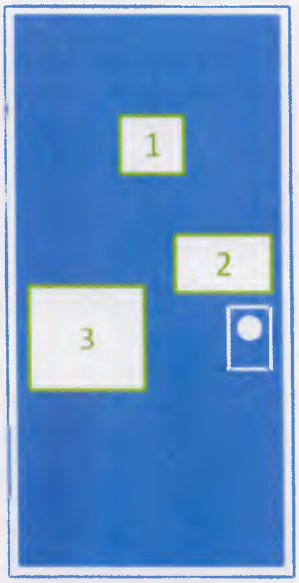


Figura 5: Diagrama de zona
Fuente: Elaboración propia

4.2.2 Implementación

Para la implementación de la puerta física, se seleccionaron los componentes específicos a utilizar para la construcción del prototipo de la puerta según la lista definida en la etapa de diseño. Se eligieron componentes económicos que

contarán con el rendimiento mínimo necesario para lograr las funcionalidades definidas. La lista completa de componentes utilizados para la construcción del prototipo puede detallarse en el Anexo H.

Figura 6: Construcción del prototipo
Fuente: Fotografía propia

Se realizaron los cortes en la puerta de madera y se fijaron los componentes en los lugares determinados. Los transformadores de energía se fijaron en el lado interno de la puerta y se unieron a una extensión para lograr que la puerta requiera de una sola fuente de energía externa. Los transformadores estarían dedicados a las cornetas, la Raspberry Pi, y a la apertura del pestillo respectivamente. Para tener mejor organización y distribución de la corriente, se colocó un cajetín que centraliza la corriente de 5V y 12V después de conectar los cables a los dispositivos correspondientes. Hay que destacar que para el cableado de los dispositivos se utilizaron colores distintos, de forma de identificar de manera visual tanto el voltaje como el dispositivo de destino.

Adicionalmente se diseñaron tres circuitos eléctricos necesarios para complementar los componentes de la puerta, siendo estos:

- Circuito de apertura de pestillo eléctrico.

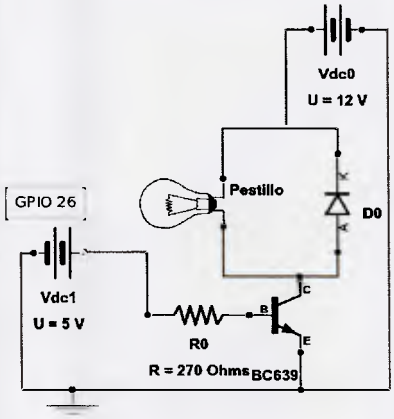


Figura 7: Circuito apertura de pestillo
Fuente: Elaboración propia

- Circuito para sensor de efecto Hall.
- Circuito de LED para iluminación en la puerta.

Para finalizar la construcción del prototipo de la puerta, se colocaron láminas de punta de diamante para proteger los componentes internos del polvo, golpes, entre otros; se instaló la puerta en un marco externo, y se ubicaron el pestillo y el imán del sensor magnético de efecto Hall.

4.2.3 Entregable

A continuación se presenta el cuadro de control con las funcionalidades completadas, y el cuadro de avance del proyecto por área al momento de finalizar esta iteración:

Funcionalidades completadas - Iteración 1	
ID	Funcionalidad
P1-1	Listar componentes
P1-2	Diseñar distribución de componentes
P1-3	Diseñar circuitos
P1-4	Construcción de circuitos
P1-5	Ensamblado de componentes

Tabla 2: Funcionalidades completadas para iteración 1. Fuente: Elaboración propia

Control de avance por área funcional					
Iteración #	Puerta	Servidor	Aplicación móvil	Dispositivos externos	Total
Iteración 1	19%	0%	0%	0%	8%
Iteración 2	-	-	-	-	-
Iteración 3	-	-	-	-	-
Iteración 4	-	-	-	-	-
Iteración 5	-	-	-	-	-

Tabla 3: Control de avance por área funcional para iteración 1. Fuente: Elaboración propia.

4.3 Iteración dos (2) - Desarrollo de software de la puerta

Durante esta iteración se diseñó y desarrolló la aplicación perteneciente al nodo de cómputo de la puerta. Es en esta aplicación en donde se implementan los métodos de autenticación y apertura automática del pestillo eléctrico, las interfaces de comunicación con el servidor central, y todas las capacidades de software necesarias para habilitar las funcionalidades de la puerta.

4.3.1 Diseño

La aplicación para el nodo de cómputo de la puerta se diseñó para contar con un mínimo grado de autonomía, esto quiere decir que sin necesidad de comunicarse con el servidor central se garantizan funcionalidades básicas del dispositivo, como lo son la autenticación y apertura de la puerta. Al contar con conexión a internet, la aplicación es capaz de comunicarse con el servidor central, habilitando la comunicación con la aplicación móvil y los dispositivos externos para el internet de las cosas.

4.3.1.1 Evaluación de tecnologías de comunicación

Un aspecto importante del diseño de la solución era la selección de los mecanismos de comunicación. Por esta razón se realizó una investigación y evaluación de tecnologías de comunicación en base a la velocidad, consumo de recursos, interoperabilidad, manejo de peticiones concurrentes, y sincronía de las peticiones. Como resultado se obtuvo el siguiente cuadro comparativo, en donde se asignaron puntuaciones del 1 al 5, en donde 5 representa el mejor puntaje. Para asignar estos valores se utilizaron como fuente distintos

trabajos de investigación que miden el rendimiento de estas tecnologías [18] [19].

	Socket internet	Servicios web	Java RMI	JMS
Seguridad	SSL	WS-Security y/o SSL	SSL	SSL
Velocidad	5	1	3	4
Consumo de recursos	5	2	4	2
Interoperabilidad	5	5	1	3
Manejo de peticiones	2	5	4	5
Síncrono	Si	Si	Si	No
Asíncrono	Si	Si	No	Si

Tabla 4: Evaluación de tecnologías de comunicación. Fuente: Elaboración propia

Como método de comunicación principal para llamadas externas hacia el nodo de cómputo se seleccionó sockets de internet, debido a su bajo consumo de recursos y velocidad, tomando en cuenta los limitados recursos de cómputo del nodo de cómputo. Mientras que para el servidor central, fue seleccionado el método de servicios web debido a su alto grado de interoperabilidad y el manejo de alto volumen de peticiones concurrentes, características importantes para lograr un alto nivel de escalabilidad en el servidor central.

4.3.2 Implementación

La aplicación principal del nodo de cómputo se desarrolló en Java SE 7.0 sobre el sistema operativo Raspbian, una distribución de Linux basada en

Debian optimizada para la Raspberry Pi. Una de las características principales de esta aplicación es la capacidad de llevar a cabo la autenticación de los usuarios aplicando métodos de análisis biométrico como reconocimiento facial y lectura de huellas dactilares.

Para el reconocimiento facial, se utilizaron las capacidades de reconocimiento facial de la librería OpenCV. Durante el proceso de registro, la aplicación realiza la captura de imágenes de una persona utilizando la cámara de la puerta. Estas imágenes son recortadas y centradas en base a la cara de la persona con el uso de una clase llamada CascadeClassifier que permite detectar el rostro y los ojos en una imagen. Estas imágenes son catalogadas para generar la llamada "base de datos facial" utilizada por la librería para generar un modelo que le permita comparar cualquier rostro con los almacenados en la base de datos, aplicando un método de análisis discriminante lineal llamado Fisherfaces [20].

Figura 8: Modelos faciales generados con Fisherfaces
Fuente: OpenCV

La Raspberry Pi cuenta con limitados recursos de cómputo, por lo que se aplicaron mejores prácticas de programación para optimizar el uso de

recursos, buscando siempre alcanzar un alto grado de eficiencia en los algoritmos desarrollados, y minimizar los hilos de procesamiento creados. Los principales algoritmos afectados fueron los relacionados con el reconocimiento facial de la librería OpenCV. En busca de optimizar el proceso de reconocimiento facial y sacar el máximo provecho del hardware disponible, se creó una aplicación, desarrollada con el lenguaje de programación C++, que es ejecutada en paralelo y lleva a cabo funciones básicas necesarias para el reconocimiento facial. Siendo C++ un lenguaje de más bajo nivel que Java, la ejecución de dichos algoritmos en ese lenguaje resultó más eficiente en términos de consumo de recursos.

Para la implementación de lectura y análisis de huellas dactilares, se utilizó libfprint, una librería de código abierto en Java que permite realizar la captura y posterior comparación de las huellas dactilares si se cuenta con el hardware necesario. Utilizando esta librería la aplicación es capaz de generar un archivo serializado único para cada huella capturada. Para mejorar la detección de huellas se capturan múltiples versiones de la misma huella por usuario, en busca de eliminar variaciones menores de posición del dedo en el lector. Para realizar la autenticación se captura una nueva huella y se busca una coincidencia contra los usuarios registrados.

La persistencia de datos de la aplicación se implementó utilizando archivos XML, siendo este un mecanismo de bajo consumo de recursos tanto de cómputo como de almacenamiento. Así, la aplicación de la puerta mantiene un registro detallado de las operaciones realizadas, como apertura, usuarios

autenticados, peticiones remotas, intentos fallidos, entre otras. El registro de eventos es respaldado en el servidor central y posteriormente eliminado de la aplicación para minimizar el espacio utilizado en el nodo de cómputo.

4.3.2.1 Evaluación de métodos de autenticación

Una vez implementados los métodos de autenticación, se procedió a realizar una evaluación de los mismos en base a velocidad de ejecución y nivel de seguridad. A continuación se presentan los resultados de dicha evaluación.

La velocidad de ejecución de los métodos de autenticación es un parámetro importante para la solución ya que impacta directamente la usabilidad de la misma. Se espera que la solución facilite el acceso a la puerta, es por esto que se realizo la medición en tiempos de ejecución para cada uno de los métodos implementados y se contrastó con el tiempo promedio de apertura de una puerta tradicional.

Métodos de autenticación	Tiempo de procesamiento (Segundos)	Proceso de autenticación completo (Segundos)
Facial	0.8	15
Huella	0.3	3
Clave	0.05	4

Tabla 5: Evaluación de métodos de autenticación según velocidad. Fuente: Elaboración propia.

Una persona tarda en promedio 8 segundos en abrir una puerta cerrada, para realizar este cálculo se simuló el proceso de apertura de una puerta con una cerradura mecánica en un ambiente real. En conclusión y según los resultados presentados en la tabla superior, existen métodos de autenticación

automatizados con bajos tiempos de ejecución que brindan una alternativa práctica y ágil a la cerradura mecánica tradicional.

Por otro lado, al evaluar la seguridad de cada método se utilizó un indicador de confianza de autenticación y una ponderación en base a la facilidad de quiebre del método. Se le asignó a cada método de autenticación un puntaje del 1 al 5, siendo 5 el puntaje mayor, y se realizó una sumatoria de los indicadores para cada combinación en busca de seleccionar la mejor en base a seguridad.

Métodos de autenticación	Confianza (5)	Quiebre (5)	Total (10)
Facial	2	2	4
Justificación: el método de reconocimiento facial implementado cuenta con un porcentaje de precisión del 70% [20], al no tener un valor alto de confianza pueden presentarse casos en donde el algoritmo confunda una persona con rasgos similares por otra. Por otro lado puede vulnerarse el método presentando una foto del usuario registrado, ya que el método realiza un análisis en 2D y no es capaz de analizar profundidad en la imagen, siendo esta una vulnerabilidad importante.			
Huella	5	5	10
Justificación: el método de reconocimiento por huella dactilar presenta una alta precisión en la lectura de huellas dactilares. Debido a la alta precisión del lector, el mismo es sensible a variaciones en la posición del dedo al escanear una huella, por lo que puede no reconocer un usuario registrado si el usuario varía la posición del dedo en el lector. Para minimizar los errores de lectura es importante capturar las huellas en múltiples posiciones. Por otro lado, al ser la huella un método de autenticación no transferible, resulta difícil de quebrar y la falsificación de una huella dactilar es un proceso complejo y poco efectivo [21].			
Clave	5	3	8
Justificación: El porcentaje de coincidencia de una clave numérica es del 100%, siendo este un método altamente confiable para la autenticación. Por otro lado una clave numérica es fácilmente transferible y pueden aplicarse métodos de fuerza bruta para quebrar el método, siendo esta la principal vulnerabilidad de las claves.			

Métodos de autenticación	Confianza (10)	Quiebre (10)	Total (20)
Facial - Huella	7	7	14
Facial - Clave	7	5	12
Huella - Clave	10	8	18
Métodos de autenticación	**Confianza (15)**	**Quiebre (15)**	**Total (30)**
Facial - Huella - Clave	12	10	22

Tabla 6: Evaluación de métodos de autenticación en base a seguridad
Fuente: Elaboración propia

Puede concluirse que por separado los métodos de autenticación aplicados presentan distintos niveles de seguridad, siendo la autenticación por huella dactilar la más segura ya que no es transferible en comparación a la clave numérica, y tiene un alto índice de confianza en contraste con el reconocimiento facial. Si bien se midió de manera individual cada uno de los métodos, se cuenta con mayor seguridad si se utiliza una combinación, siendo la mejor combinación de dos métodos el uso de huella dactilar y clave numérica. Por otro lado, si se requiere del mayor grado de seguridad en la puerta la mejor opción es aplicar los tres métodos de autenticación en conjunto, aumentando la confianza del proceso y minimizando el riesgo de quiebre o autenticación forzada en la puerta.

4.3.3 Entregable

A continuación se presenta el cuadro de control con las funcionalidades completadas, y el cuadro de avance del proyecto por área al momento de finalizar esta iteración:

Funcionalidades completadas - Iteración 2	
ID	Funcionalidad
P2-1	Validar mediante una clave numérica la identidad de un usuario
P3-1	Validar mediante detección de huella dactilar la identidad de un usuario
P4-1	Validar mediante detección facial la identidad de un usuario
P4-2	Encender una luz para mejorar iluminación en reconocimiento facial
P4-3	Enviar foto de cámara integrada en la puerta al servidor
P5-1	Integrar los distintos métodos de autenticación para abrir el pestillo de la puerta
P5-2	Implementar menú de usuario utilizando sonidos y teclado
P6-1	Abrir pestillo de la puerta
P7-1	Emitir sonido de timbre
P9-1	Gestión de usuarios
P11-1	Persistencia de la información de configuración
P11-2	Registro de eventos de la puerta
P11-3	Persistencia de eventos en puerta

Tabla 7: Funcionalidades completadas para iteración 2. Fuente: Elaboración propia

Control de avance por área funcional					
Iteración #	Puerta	Servidor	Aplicación móvil	Dispositivos externos	Total
Iteración 1	19%	0%	0%	0%	8%
Iteración 2	69%	0%	0%	0%	31%
Iteración 3	-	-	-	-	-
Iteración 4	-	-	-	-	-
Iteración 5	-	-	-	-	-

Tabla 8: Control de avance por área funcional para iteración 2. Fuente: Elaboración propia

4.4 Iteración tres (3) - Desarrollo de servidor central

El servidor central es un componente fundamental de la arquitectura ya que aquí se centraliza la comunicación entre las puertas, los dispositivos móviles, y los dispositivos externos del internet de las cosas. Al mismo tiempo que almacena los datos de perfiles de usuario, puertas registradas, y dispositivos externos sincronizados.

4.4.1 Diseño

Debido a la naturaleza centralizada de la arquitectura, el servidor central requería lograr un alto grado de seguridad, interoperabilidad y escalabilidad. En la actualidad los modelos de computación en la nube brindan una gran cantidad de ventajas ya que ofrecen un alto grado de seguridad y escalabilidad, con características como la gestión y detección proactiva de ataques, o capacidades de asignación dinámica de recursos de cómputo en base a la demanda, entre otros.

Por esta razón se diseñó el servidor central siguiendo mejores prácticas que permitieran la migración hacia un modelo de computación en la nube de manera sencilla, al mismo tiempo que cubriera los requerimientos funcionales del proyecto. Los componentes principales del servidor central son:

- **Servidor VPN:** para lograr una conexión segura y realizar el enlace con las puertas sin dirección IP pública, se decidió utilizar un servidor VPN el cual cifra el canal de comunicación entre el servidor central y el

nodo de cómputo presente en la puerta, al mismo tiempo que facilita el monitoreo y gestión de los dispositivos conectados a la red.

- **Servidor de aplicaciones:** este componente es utilizado para gestionar configuraciones de seguridad y credenciales de acceso para la capa de aplicación, además de administrar la seguridad y acceso a APIs públicos.
- **Aplicación de servidor central:** en esta aplicación es implementada la lógica necesaria para enlazar y orquestar las interacciones entre los componentes de la arquitectura.
- **Manejador de base de datos:** si bien la mayor parte de la transaccionalidad es automatizada, también se espera que los usuarios modifiquen datos de manera concurrente. Por esta razón se seleccionó el uso de un manejador de base de datos relacional para proveer la persistencia de datos de la aplicación de servidor central.

4.4.2 Implementación

Para dedicar los recursos necesarios y encapsular las aplicaciones instaladas, se utilizó una máquina virtual Oracle VM VirtualBox con sistema operativo Ubuntu 14.4. Como servidor VPN fue utilizado OpenVPN, siendo este compatible con el cliente instalado en el prototipo de la puerta. Para garantizar la persistencia de datos se implementó el manejador de base de datos PostgreSQL 9.4, sobre el cual se desplegó la base de datos relacional diseñada para la aplicación de servidor central; el modelo entidad-relación de la base de datos puede detallarse en el Anexo D. El servidor de aplicaciones

instalado y configurado fue GlassFish Server 4.1, sobre el cual se desplegó la aplicación de servidor central, desarrollada con Java EE 7.

La aplicación de servidor central implementada consta de cuatro módulos:

1. **Módulo de comunicación para puertas:** para lograr la comunicación con el nodo de cómputo de las puertas, el servidor central hace uso de servicios web para interacciones puerta-servidor, y de sockets de internet para interacciones servidor-puerta.

2. **Módulo de servicios para dispositivos móviles:** este módulo expone un API de servicios web utilizados para realizar la comunicación entre el servidor central y los dispositivos móviles.

3. **Módulo de integración con dispositivos externos:** una característica importante del servidor central es la de dotar de interoperabilidad a la solución de puerta inteligente, por lo que fue diseñado para facilitar la integración de nuevos dispositivos externos. En este módulo se realizan las implementaciones para cada nuevo dispositivo externo a integrar con la solución.

4. **Módulo central:** este módulo es el encargado de realizar validaciones de seguridad globales en la aplicación de servidor central.

En la figura a continuación puede apreciarse un diagrama que muestra los componentes utilizados para el despliegue del servidor central.

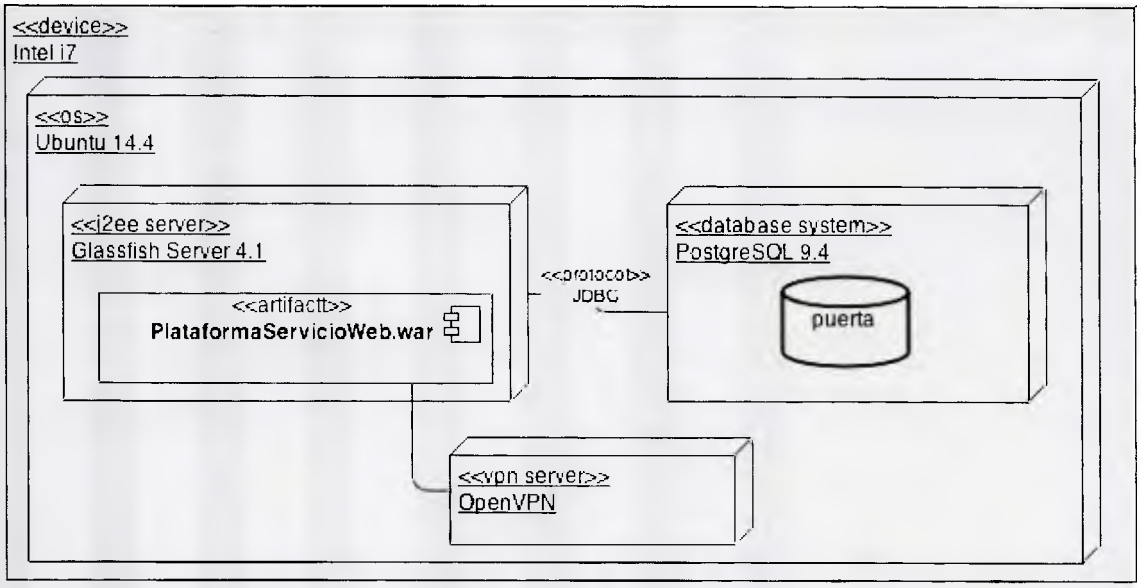


Figura 9: Diagrama de componentes para servidor central
Fuente: Elaboración propia

4.4.3 Entregable

A continuación se presenta el cuadro de control con las funcionalidades completadas, y el cuadro de avance del proyecto por área al momento de finalizar esta iteración:

Funcionalidades completadas - Iteración 3	
ID	Funcionalidad
P7-2	Disparar proceso de timbre inteligente
P12-1	Conexión por socket al servidor central
P12-2	Conexión a VPN del servidor central
P12-3	Envío recurrente de respaldo del registro de eventos a servidor
S1-1	Exposición de servicios web para puerta
S1-2	Envío de peticiones por socket a puerta
S2-1	Exposición de servicios web para aplicación móvil
S4-1	Exposición en red pública de servicios
S4-2	Habilitar únicamente conexión SSL a servicios
S4-3	Despliegue de aplicación de servidor central
S5-1	Persistencia de datos en BD
S6-1	Inicializar red segura VPN
S6-2	Recepción de conexiones por red VPN

S6-3	Asignación de IPs a puertas
S7-1	Realizar validaciones de control de acceso
S7-2	Enrutamiento de peticiones

Tabla 9: Funcionalidades completadas para iteración 3. Fuente: Elaboración propia

Control de avance por área funcional					
Iteración #	Puerta	Servidor	Aplicación móvil	Dispositivos externos	Total
Iteración 1	19%	0%	0%	0%	8%
Iteración 2	69%	0%	0%	0%	31%
Iteración 3	85%	79%	0%	0%	56%
Iteración 4	-	-	-	-	-
Iteración 5	-	-	-	-	-

Tabla 10: Control de avance por área funcional para iteración 3. Fuente: Elaboración propia

4.5 Iteración cuatro (4) - Desarrollo de aplicación móvil

El foco de esta iteración estuvo en la creación de la aplicación móvil, cuya función es la de actuar como interfaz principal entre la puerta y el usuario.

4.5.1 Diseño

La plataforma móvil Android fue la seleccionada para el desarrollo de la aplicación en base a su madurez técnica y apertura a nivel de sistema operativo, brindando una gran cantidad de posibilidades de desarrollo e integración.

Siendo este el único componente de software con interfaz gráfica en la solución, se hizo énfasis en diseñar una buena experiencia de usuario. Para lograr este objetivo se siguieron las mejores prácticas de diseño asociadas a la plataforma de desarrollo móvil seleccionada. Android cuenta con detallados lineamientos y manuales de diseño de interfaces móviles, los cuales fueron estudiados y aplicados.

Se construyeron modelos de maquetación para las pantallas de la aplicación, como se muestra en el Anexo F. Estas maquetas se utilizaron para diseñar y conceptualizar el flujo de navegación y aspecto general de la aplicación antes de desarrollarla. Utilizando este tipo de modelos pueden detectarse errores de diseño general de manera temprana, ahorrando tiempo valioso durante la etapa de desarrollo. En la figura presentada a continuación pueden observarse las maquetas realizadas para la pantalla de inicio y el menú lateral.

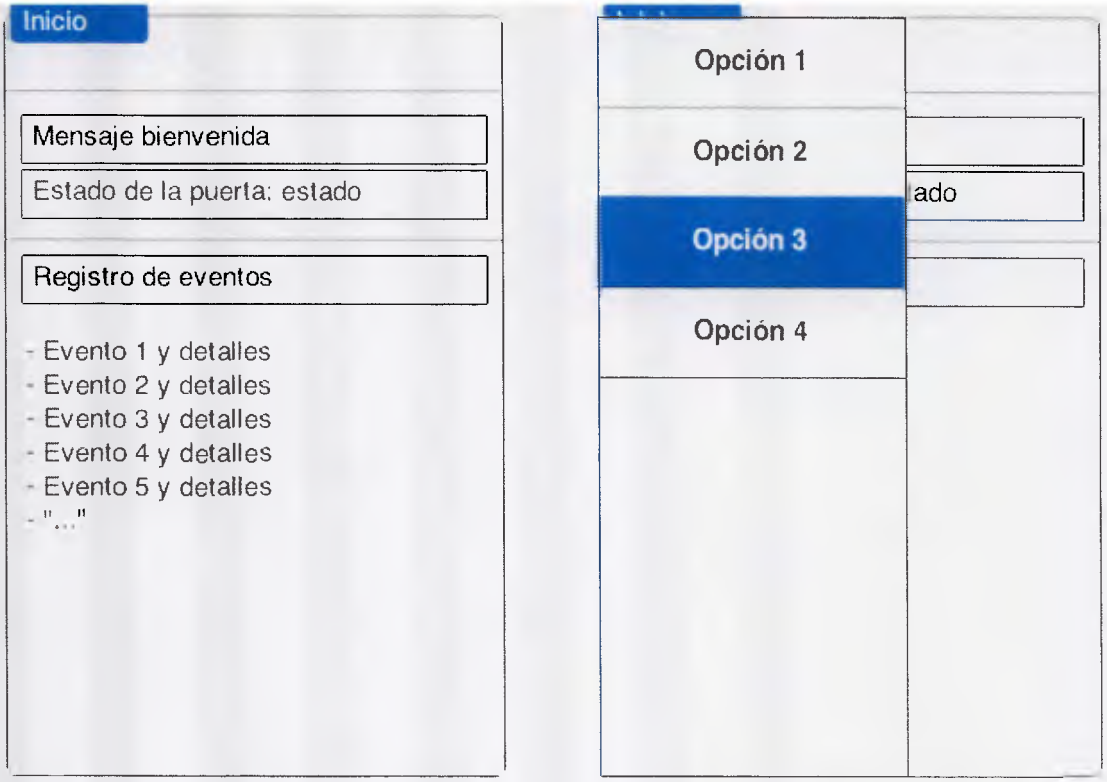


Figura 10: Maquetas de interfaz para aplicación móvil
Fuente: Elaboración propia

Una de las funcionalidades más complejas implementadas durante esta iteración fue la de registro de nuevo usuario, ya que requería de la coordinación de todos los componentes de la solución, comenzando por el dispositivo móvil y terminando con la puerta. El proceso de registro de nuevo usuario es iniciado mediante la aplicación móvil, la cual le permite al usuario ingresar su información básica y entrega un código de registro para iniciar el proceso de registro en la puerta. Este código de registro tiene una validez temporal y es utilizado para aparear la puerta con la cuenta del usuario. Una vez introducido el código, la puerta realiza una validación del mismo con el servidor central, y de ser válido inicia el proceso de registro. Al finalizar el proceso se envía un correo electrónico al usuario para informarle si el proceso de registro fue completado con éxito. En el modelo de secuencia de sistemas

presentado a continuación puede apreciarse el flujo completo del proceso de registro diseñado.

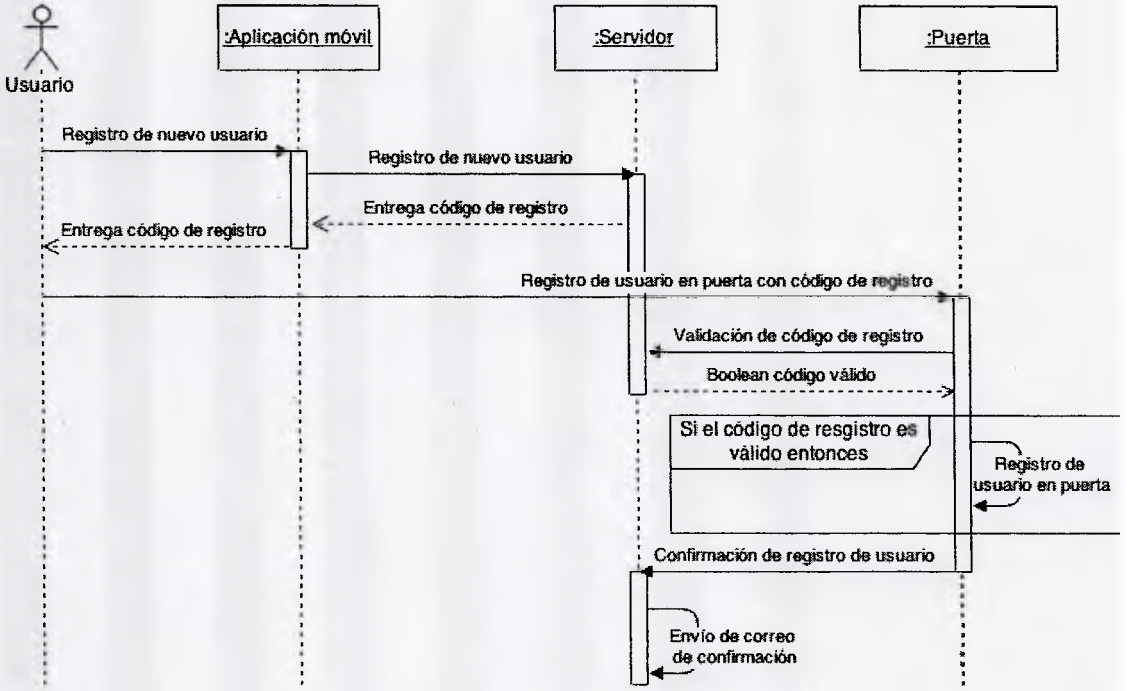


Figura 11: Diagrama de secuencia para proceso de registro
Fuente: Elaboración propia

4.5.2 Implementación

Durante la fase de desarrollo se trabajó con el IDE Android Studio, actualmente entorno de desarrollo oficial para Android. El lenguaje de programación utilizado para la implementación fue Java.

La mayor parte de las funcionalidades de la aplicación móvil son dependientes del servidor central, el cual provee un API de servicios web REST que funciona como mediador entre la puerta y el dispositivo móvil.

4.5.3 Entregable

A continuación se presenta el cuadro de control con las funcionalidades completadas, y el cuadro de avance del proyecto por área al momento de finalizar esta iteración:

Funcionalidades completadas - Iteración 4	
ID	Funcionalidad
M1-1	Solicitar estado de la puerta
M2-1	Vista remota de una puerta
M2-2	Permitir al usuario abrir la puerta de manera remota
M3-1	Permitir al usuario disparar proceso de pánico
M4-1	Permitir al usuario ver historial de eventos recientes en la puerta
M5-1	Ingresar a la aplicación móvil un usuario
M5-2	Registrar en la aplicación móvil un usuario
M5-3	Editar información básica de cuenta de usuario
M5-4	Eliminar usuario de aplicación y puerta
M7-1	Validar con contraseña de la puerta para apertura remota
M7-2	Configurar correos electrónicos de usuarios para botón de pánico
M7-3	Configurar métodos de acceso activos en puerta
P8-1	Emitir sonido de alarma
P8-2	Disparar proceso de alarma
P10-1	Configurar métodos de autenticación activos

Tabla 11: Funcionalidades completadas para iteración 4. Fuente: Elaboración propia

Control de avance por área funcional					
Iteración #	**Puerta**	Servidor	Aplicación móvil	Dispositivos externos	Total
Iteración 1	19%	0%	0%	0%	8%
Iteración 2	69%	0%	0%	0%	31%
Iteración 3	85%	79%	0%	0%	56%
Iteración 4	96%	86%	92%	0%	83%
Iteración 5	-	-	-	-	-

Tabla 12: Control de avance por área funcional para iteración 4. Fuente: Elaboración propia

4.6 Iteración cinco (5) - Desarrollo de dispositivos externos

Durante esta iteración se diseñaron e implementaron en el servidor central las características técnicas relacionadas al internet de las cosas, como también el desarrollo de dos simuladores de dispositivo externo, los cuales son integrados con la solución para habilitar nuevas funcionalidades.

4.6.1 Diseño

Para lograr la flexibilidad necesaria en el modelo lógico de la solución, se diseñó una clase abstracta de dispositivo que representa los atributos y métodos base necesarios para implementar un dispositivo externo, y crear una capa de abstracción que facilita la integración de nuevos dispositivos en el futuro. De la misma manera se diseñó la capa de persistencia de datos, para soportar dicha flexibilidad utilizando un modelo relacional que implementa columnas de documento JSON que facilitan el mantenimiento y extensión de la base de datos a la hora de implementar nuevos dispositivos en la capa de aplicación. El diagrama de clases y modelo entidad-relación del servidor central puede detallarse en el Anexo C y el Anexo D.

Por otro lado, uno de los objetivos del proyecto consta de la ejecución de una prueba de concepto que demuestre escenarios de prueba cercanos a la realidad, para de ilustrar las capacidades funcionales de la solución como también la aplicabilidad del concepto del internet de las cosas. Para diseñar dichos escenarios era necesario contar con dispositivos externos con capacidades de comunicación por internet que permitieran ser integrados con

la solución de puerta inteligente extendiendo sus funcionalidades. En base a esta premisa se identificó la necesidad de construir y desarrollar dos simuladores de dispositivo externo, estos son:

1. Lámpara con encendido remoto.
2. Reproductor de música remoto.

4.6.2 Implementación

Durante esta iteración se desarrollaron las capacidades de la solución relacionadas con el internet de las cosas. De la misma manera, se desarrollaron y construyeron los simuladores de dispositivo externo con capacidades de comunicación por internet.

Para la construcción de la lámpara se utilizó un relé de dos canales, conectado a dos tomas de corriente y a los pines de la Raspberry Pi. Para activar cada toma de corriente eléctrica, desde el pin de la Raspberry Pi se emite una señal y se activa el mismo. Para que funcione como un dispositivo del internet de las cosas se desarrolló una aplicación en Java que permite realizar el encendido remoto de la lámpara mediante internet utilizando servicios web.

Para el reproductor de música se desarrolló una aplicación en Java, que recibe peticiones desde internet utilizando servicios web, que permite la reproducción remota de archivos MP3. Las peticiones contienen los datos del usuario, y el nombre y ruta del archivo a reproducir.

Una vez construidos los simuladores de dispositivo externo, se realizó la integración con la solución de puerta inteligente, extendiendo el modelo

lógico y de persistencia, para de esta manera poner a prueba la flexibilidad de estos en la práctica.

4.6.3 Entregable

A continuación se presenta el cuadro de control con las funcionalidades completadas, y el cuadro de avance del proyecto por área al momento de finalizar esta iteración:

Funcionalidades co[illegible] - [illegible]ación 5	
ID	Funcionalidad
D1-1	Ensamblar componentes
D2-1	Encender lámpara de manera automatizada
D2-2	Exponer servicio para encender lámpara de manera remota
D3-1	Reproducir MP3 de manera automatizada
D3-2	Exponer servicio para reproducción de MP3 de manera remota
D4-1	Enviar correos electrónicos de manera automatizada
M6-1	Gestionar dispositivos conectados a la puerta
P10-2	Creación de script de arranque de la puerta
S3-1	Integración con lámpara
S3-2	Integración con reproductor de música

Tabla 13: Funcionalidades completadas para iteración 5. Fuente: Elaboración propia

Control de avance por área funcional					
Iteración #	Puerta	Servidor	Aplicación móvil	Dispositivos externos	Total
Iteración 1	19%	0%	0%	0%	8%
Iteración 2	69%	0%	0%	0%	31%
Iteración 3	85%	79%	0%	0%	56%
Iteración 4	96%	86%	92%	0%	83%
Iteración 5	100%	100%	100%	100%	100%

Tabla 14: Control de avance por área funcional para iteración 5. Fuente: Elaboración propia

4.7 Pruebas de concepto

Una vez concluido el proyecto se realizó el ejercicio de prueba de concepto, en el que se ejecutaron tres escenarios de demostración utilizando como usuarios a los miembros del equipo de este trabajo de grado. Estos escenarios buscan demostrar la aplicabilidad y funcionamiento general de la solución de forma práctica, así como también encontrar puntos de mejora en la solución lograda enfrentando escenarios reales de uso.

La solución construida fue desplegada y utilizada de manera continua sin interrupciones durante un período de dos semanas, en el cual se evaluó la estabilidad de los componentes. Durante estas dos semanas se realizaron de manera repetida tres escenarios de uso, descritos en detalle a continuación:

- **Escenario 1:** al realizar con éxito el proceso de autenticación y apertura de la puerta se enciende de manera automática el simulador de lámpara.
- **Escenario 2:** al realizar con éxito el proceso de autenticación y apertura de la puerta se reproduce una canción preseleccionada haciendo uso del simulador de reproductor de música.
- **Escenario 3:** al realizar con éxito el proceso de autenticación y apertura de la puerta es enviado un correo electrónico de manera automática con contenido y dirección de destino pre-configurados.

Cada uno de los flujos presentados en estos escenarios fueron completados con éxito y arrojaron información valiosa que permitió al equipo identificar áreas de mejora y corregir errores menores de la solución.

Luego de finalizadas las correcciones necesarias, identificadas durante las pruebas de concepto, se completó el ciclo de vida del proyecto definido en el alcance de este trabajo de grado de forma satisfactoria, resultando en un producto de software y hardware con todas las características funcionales esperadas.

5.1 Arquitectura de solución para el internet de las cosas

A continuación se describe la arquitectura del producto resultante haciendo énfasis en las decisiones de diseño tomadas por el equipo para superar los principales retos técnicos asociados al internet de las cosas, descritos en el apartado 2.3 Retos técnicos asociados al internet de las cosas.

Se diseñó una arquitectura orientada a servicios que está compuesta por tres componentes principales, estos son las puertas inteligentes, servidor central, y dispositivos móviles. En el diagrama presentado a continuación pueden detallarse las tecnologías utilizadas para realizar el despliegue de la arquitectura, como también los mecanismos de utilizados.

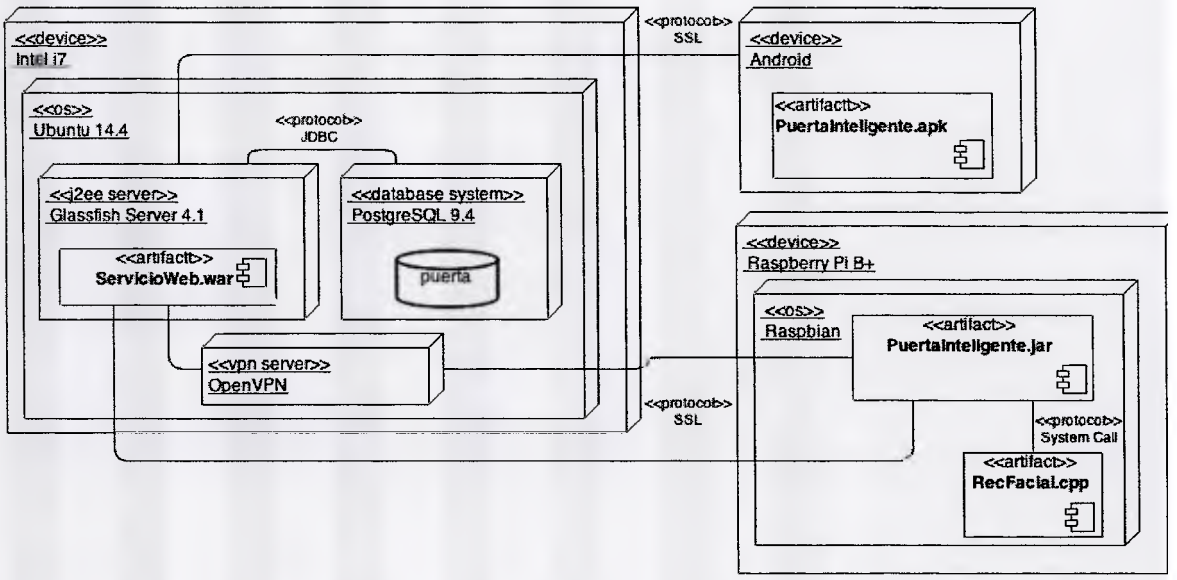


Figura 12: Diagrama de despliegue de la solución
Fuente: Elaboración propia

La arquitectura de solución fue diseñada para lograr un alto grado de interoperabilidad, escalabilidad, autonomía del dispositivo, seguridad y privacidad para superar los retos técnicos principales que enfrentan las soluciones diseñadas para el internet de las cosas. A continuación se describen las decisiones de diseño tomadas por el equipo para lograr las características mencionadas.

5.1.1 Interoperabilidad

Uno de los retos al diseñar un dispositivo para el internet de las cosas, es el de lograr una arquitectura que facilite la interoperabilidad con dispositivos externos. La arquitectura de la puerta inteligente fue diseñada con la intención de que el único punto de integración entre la puerta y los dispositivos externos sea el servidor central. De esta manera no es necesario actualizar o modificar el software instalado en el nodo de cómputo de la puerta, ya que el servidor central es el que implementa los métodos de comunicación con nuevos dispositivos, y adicionalmente orquesta las interacciones entre ambos, logrando así una capa de abstracción entre las puertas y los dispositivos externos. Esta capa de abstracción facilita los esfuerzos de integración de la solución y habilita la interoperabilidad de la arquitectura.

Otra característica importante de la solución es la flexibilidad y facilidad de integración lograda. La solución fue diseñada para minimizar el impacto y facilitar los esfuerzos de integración a la hora de integrar un nuevo dispositivo externo del internet de las cosas. La estrategia de integración diseñada para la

solución se basa en implementar una nueva clase en el servidor central, que extienda del modelo lógico de la clase "dispositivo", esta clase brinda una capa de abstracción que permite a la solución gestionar dispositivos sin conocer los detalles del mismo. Por otro lado, el modelo relacional de la base de datos del servidor cuenta con columnas JSON que soportan el modelo abstracto de dominio sin necesidad de realizar modificaciones en el modelo relacional. De esta manera se logro diseñar un modelo lógico que soporta un alto nivel de flexibilidad y mantenimiento, al mismo tiempo que se garantiza la consistencia de los datos utilizando una base de datos relacional lo suficientemente flexible para soportar el modelo lógico diseñado. En el diagrama de clases a continuación puede detallarse la extensión del modelo lógico al implementar nuevas clases para los dispositivos externos del reproductor y lámpara.

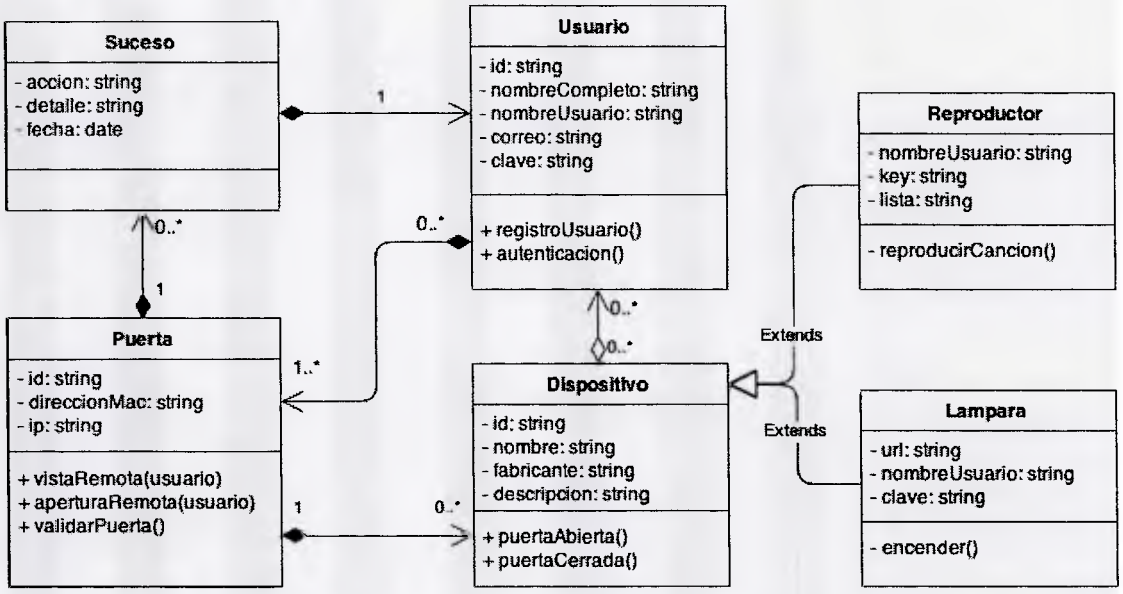


Figura 13: Diagrama de clases para servidor central
Fuente: Elaboración propia

Esta integración con dispositivos externos también expande la arquitectura inicial de la solución, en el diagrama de despliegue a continuación se puede observar el impacto de dicha integración en la arquitectura original.

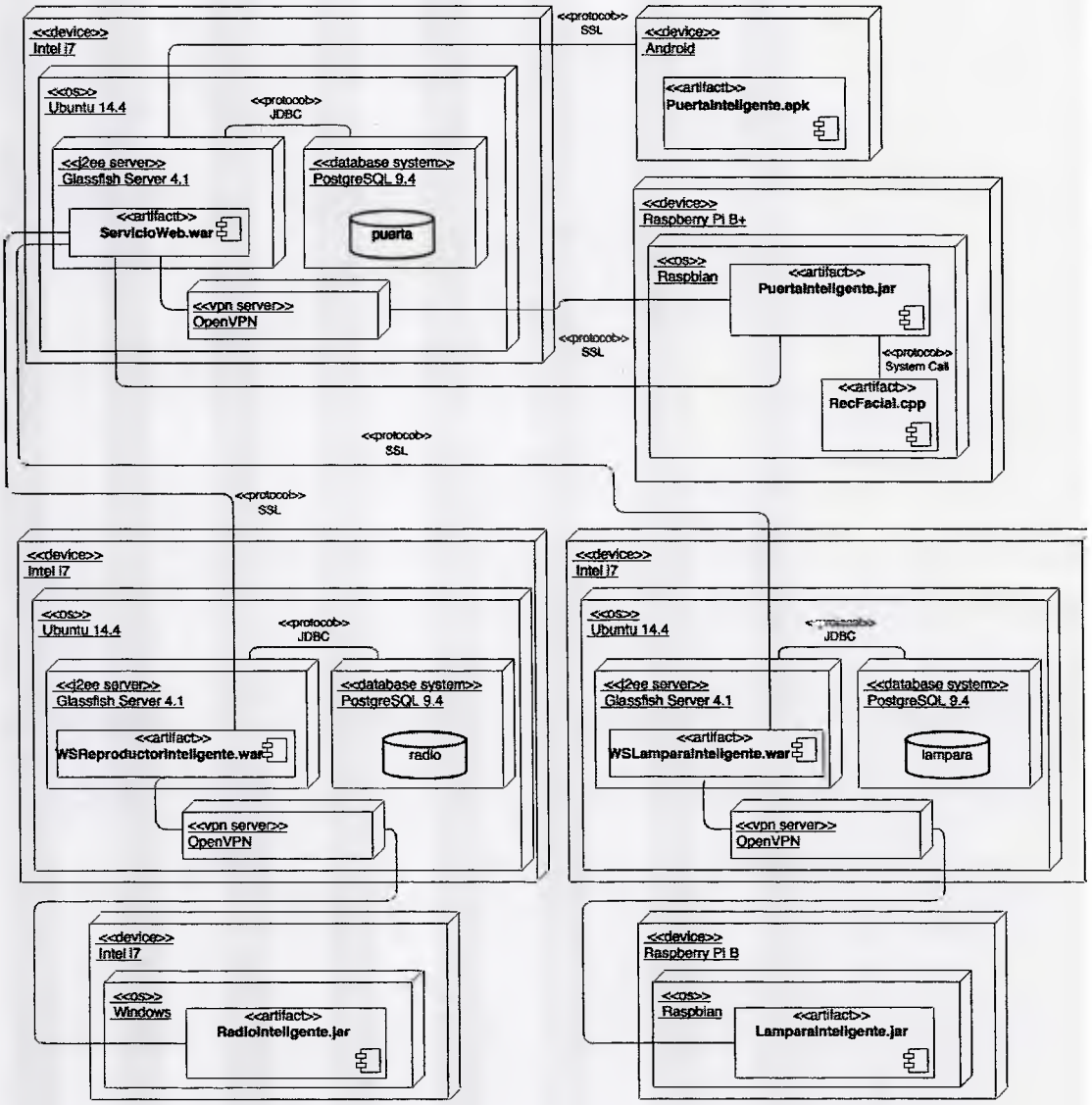


Figura 14: Arquitectura de la solución con dispositivos externos
Fuente: Elaboración propia

5.1.2 Escalabilidad

Otro aspecto clave para una arquitectura del internet de las cosas es que sea escalable, ya que la naturaleza de la misma tiende a aumentar en el número de dispositivos gestionados a lo largo del tiempo.

La arquitectura diseñada cuenta con un modelo centralizado de comunicación; por esta razón se hizo énfasis seleccionar una tecnología de comunicación que brindara alta escalabilidad para el servidor central, siendo este el punto más vulnerable de saturación en la arquitectura.

En el apartado 4.3.1.1 Evaluación de tecnologías de comunicación, puede detallarse la selección y evaluación realizada, como también la selección de servicios web como principal método de comunicación del servidor con los demás componentes de la arquitectura, en base a su eficiencia al manejar un alto volumen de peticiones concurrentes. Las puertas hacen uso de conexiones por socket para recibir peticiones desde el servidor central, esto debido a su eficiencia y bajo consumo de recursos en modo servidor, ya que implementar un servidor web en el nodo de cómputo de la puerta para recibir peticiones de servicios web demandaba gran cantidad de recursos de la misma, necesarios para realizar las operaciones de autenticación.

En el diagrama de componentes a continuación pueden detallarse las tecnologías de comunicación utilizadas en la arquitectura.

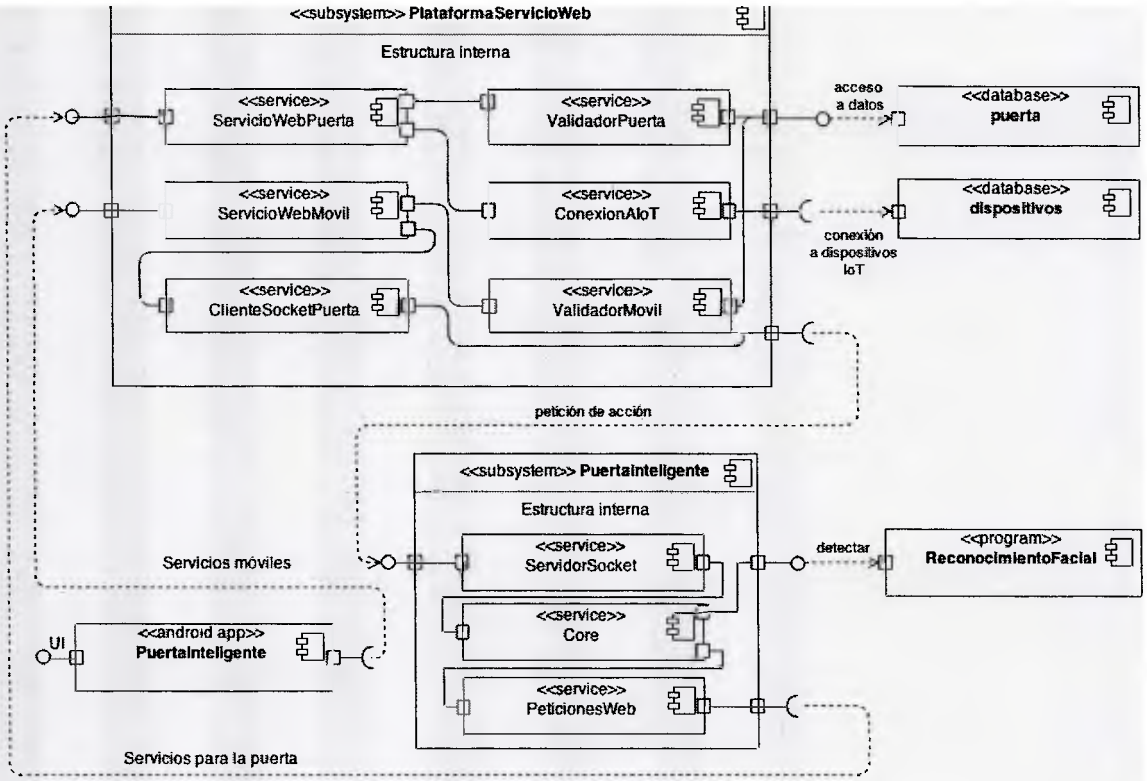


Figura 15: Diagrama de componentes para servidor central
Fuente: Elaboración propia

Adicionalmente se implementó el servidor central utilizando una plataforma tecnológica compatible con la nube que permitiera su migración a dicho esquema de manera sencilla. Estando el servidor central alojado en la nube, la arquitectura puede tomar provecho de las capacidades de elasticidad en este tipo de plataformas, lo cual le permitirá al servidor central gestionar los recursos disponibles en base a la demanda, quedando así minimizados los riesgos de caída y saturación.

5.1.3 Autonomía del dispositivo

Dependiendo de la solución, muchos de los dispositivos diseñados para el internet de las cosas cuentan con un grado de autonomía, lo cual le permite al usuario hacer uso de ciertas características funcionales sin necesitar de una conexión a internet; pero en muchas ocasiones se diseñan dispositivos que hacen un fuerte uso de arquitecturas cliente-servidor para habilitar la mayoría de sus funcionalidades, en busca de abaratar costos minimizando el procesamiento presente en el mismo y delegando tareas al servidor. Por otro lado, esto implica una pérdida de autonomía en el dispositivo, en donde al carecer de conexión a internet el mismo pierde la capacidad de operar.

La puerta inteligente construida en este trabajo de grado fue diseñada para contar con un grado de autonomía básica sin necesidad de internet. Para lograrlo se integró un nodo de cómputo dentro del dispositivo con los recursos de cómputo necesarios para habilitar características funcionales básicas relacionadas con la seguridad en la puerta, como lo son la autenticación y apertura automática para el usuario final. De esta manera, el dispositivo no requiere de una conexión a internet para operar, sino que hace uso de la misma para extender sus capacidades. Esta autonomía tiene un alto impacto en la experiencia del usuario con el dispositivo, al garantizar un mínimo de funcionamiento en todo momento sin necesidad de internet.

5.1.4 Seguridad y privacidad

Uno de los aspectos más delicados al diseñar arquitecturas para el internet de las cosas, es la seguridad y privacidad de los datos del usuario. Por naturaleza este tipo de arquitecturas son distribuidas y orientadas a servicios, al mismo tiempo que hacen uso extenso de internet y APIs públicas para habilitar la comunicación con dispositivos externos.

Una de estas decisiones está relacionada a la gestión de datos sensibles. Todos los datos utilizados para la autenticación de los usuarios en la puerta son considerados sensibles, y son almacenados y gestionados únicamente por el nodo de cómputo presente en la puerta. Dentro del mismo se encuentran las fotos, huellas y claves numéricas de manera cifrada bajo dm-crypt/LUKS. No existen métodos remotos para la gestión de estos datos, por lo que el servidor central no puede consultar, modificar, ni eliminar los mismos.

En cuanto a la autenticación y apertura de la puerta, el proceso es ejecutado por completo de manera local, sin intervención del servidor central. En caso de presentarse una petición de apertura remota desde el dispositivo móvil, el usuario debe introducir su contraseña de acceso numérica y esta es enviada utilizando como puente el servidor central, el cual entrega la contraseña al nodo de cómputo presente en la puerta en donde se realiza el proceso de autenticación y apertura de manera local, como se muestra en el diagrama de secuencia de sistemas a continuación.

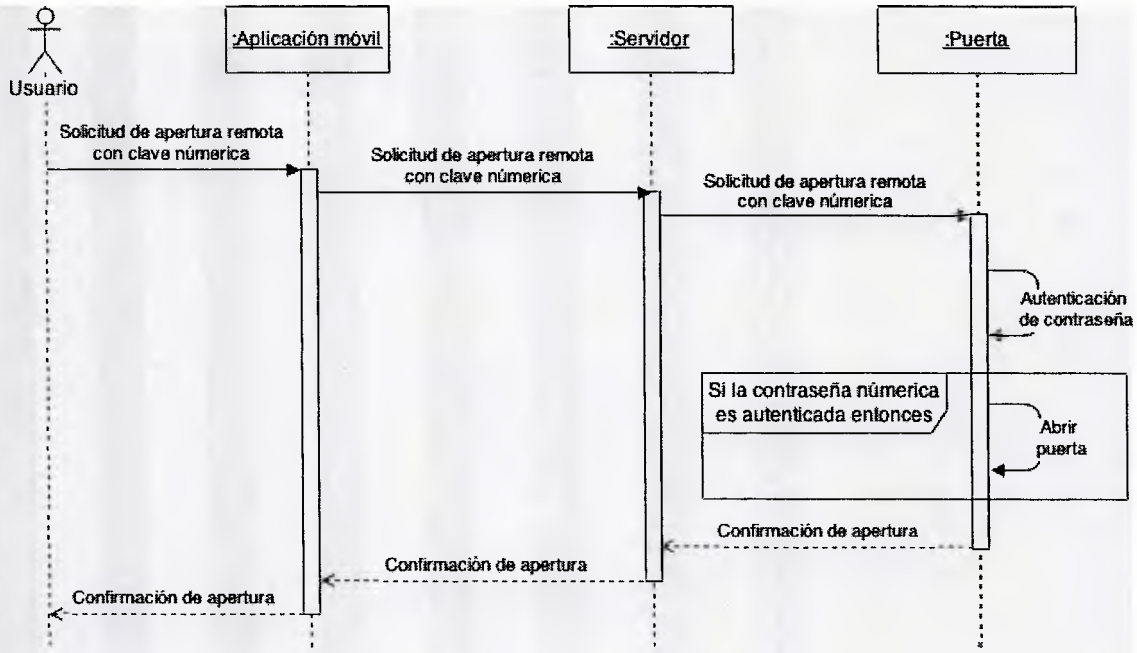


Figura 16: Diagrama de secuencia para apertura remota
Fuente: Elaboración propia

De esta manera se encapsula tanto la gestión de datos sensibles como el proceso de autenticación y apertura dentro del nodo de cómputo presente en la puerta minimizando el impacto de una brecha de seguridad en el servidor central, ya que desde el mismo no puede realizarse ninguna operación de autenticación, apertura remota o acceso a los datos sensibles de los usuarios.

El resultado final de este trabajo de grado cumplió con los objetivos y el alcance definido. En este apartado se describen las características funcionales del producto resultante, como también la cobertura de los objetivos definidos.

6.1 Puerta inteligente para el internet de las cosas

La solución está compuesta por componentes de software y hardware entre los cuales se encuentran el prototipo de puerta, la aplicación móvil, el servidor central y los simuladores de dispositivo externo. Las funcionalidades de la misma van desde mejoras de seguridad y control remoto de la puerta, hasta interacciones automáticas entre la puerta y los simuladores de dispositivo externo construidos, como la lámpara y el reproductor de música.

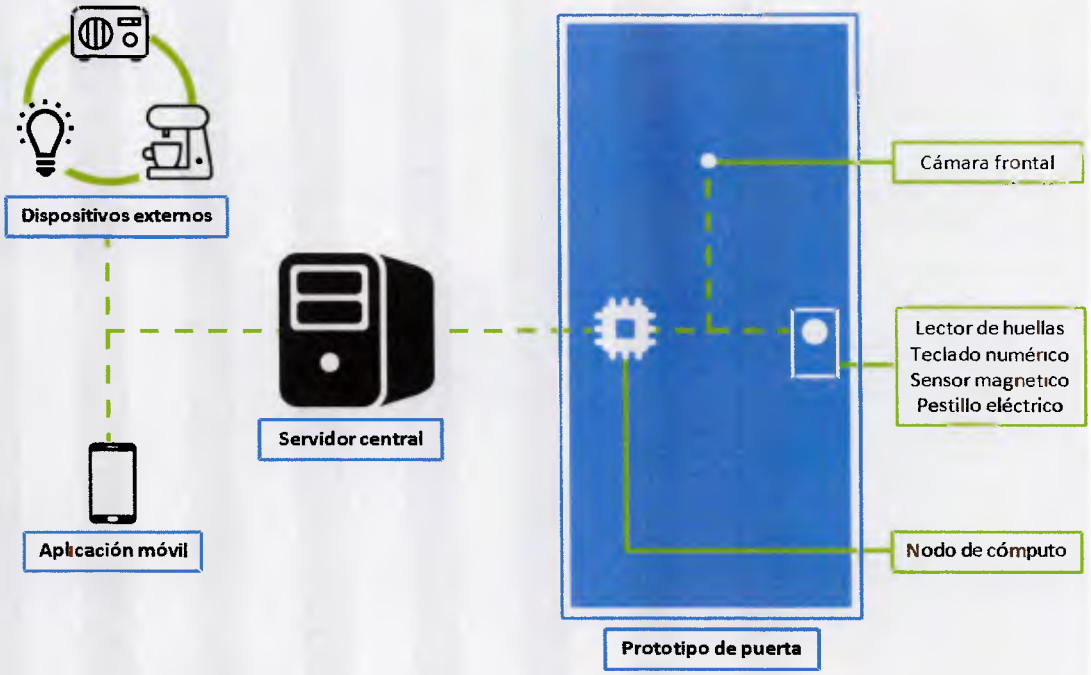


Figura 17: Modelo general del producto resultante. Fuente: Elaboración propia

6.2 Prototipo de puerta

En cuanto a las mejoras de seguridad, el prototipo de puerta construido es capaz de aplicar tres métodos de autenticación distintos: reconocimiento facial, detección de huella dactilar, e identificación por clave numérica. Estos métodos de autenticación pueden ser activados o desactivados a conveniencia del usuario.

La puerta no cuenta con una interfaz gráfica; esta hace uso de una interfaz combinada por entradas de teclado y audio generado por computadora que guía al usuario al interactuar con la puerta de manera directa. Desde esta interfaz pueden realizarse las siguientes acciones:

- **Autenticación y apertura automatizada de pestillo eléctrico:** al marcar la tecla más (+) del teclado numérico, se inicia el proceso de autenticación con los métodos activos en la puerta y de ser exitoso se realiza la apertura automática del pestillo eléctrico. Esta funcionalidad no requiere de conexión a internet.
- **Inicio de proceso de registro para nuevos usuarios:** al marcar la tecla menos (-) del teclado numérico, se inicia el proceso de registro de nuevo usuario, el cual requiere de un código generado por la aplicación móvil a través del cual se enlaza la cuenta creada en el servidor central con la puerta en donde se efectúa el registro. Durante este proceso se toman las fotos, huellas y contraseña necesarias para realizar la autenticación del usuario. Esta funcionalidad requiere de conexión a internet.

- **Timbre inteligente:** al marcar la tecla asterisco (*) del teclado numérico, es enviado un correo electrónico con la foto de la cámara frontal a todos los usuarios registrados en la puerta. Esta funcionalidad requiere de conexión a internet.

Es importante destacar que las funcionalidades básicas de la puerta tales como autenticación, apertura automática del pestillo, y registro, pueden ser utilizadas sin contar con una conexión a internet. De esta manera se garantiza un mínimo de funcionalidad y autonomía del dispositivo incluso si se pierde la conectividad.

Figura 13: Prototipo de puerta
Fuente: Fotografía propia

6.3 Aplicación móvil

Adicionalmente la solución cuenta con una aplicación para dispositivos móviles inteligentes con el sistema operativo Android 3.0 o superior, que opera como la interfaz gráfica principal entre la puerta y el usuario. Esta aplicación cuenta con funcionalidades varias, desde la gestión de la información básica del usuario, hasta el control remoto de la puerta. A continuación se describen las funcionalidades de la aplicación:

Figura 19: Pantalla de inicio
Fuente: Elaboración propia

- **Registro de nuevos usuarios:** a través de la aplicación móvil el usuario tiene la capacidad de solicitar un código de registro de nuevo usuario para iniciar el proceso en la puerta.

- **Consulta de estado de la puerta:** desde la aplicación móvil el usuario puede consultar el estado actual de la puerta, si esta se encuentra cerrada o abierta.

- **Vista remota de la puerta:** el usuario puede solicitar una foto de manera remota en tiempo real y observar todo lo que esté dentro del campo de visión de la cámara instalada en la puerta.

- **Apertura remota de la puerta:** el usuario puede hacer uso de esta funcionalidad para aperturar el pestillo eléctrico de la puerta de manera remota. Para realizar esta operación el usuario debe ingresar su clave

numérica ya que por razones de seguridad el servidor central no posee la capacidad de abrir las puertas registradas de manera directa.

- **Botón de pánico:** en caso de presentarse una emergencia, el usuario puede iniciar un protocolo de emergencia que consiste en la emisión de un sonido de alarma desde la puerta y el envío de un correo electrónico de alerta a los contactos de emergencia del usuario.

- **Configuración de puerta:** desde la aplicación móvil pueden activarse y desactivarse métodos de autenticación a preferencia del usuario.

- **Configuración de usuario:** dentro de las configuraciones de usuario puede gestionarse la información básica del usuario.

- **Gestión de dispositivos conectados:** desde la aplicación móvil pueden activarse y desactivarse dispositivos externos, y configurar sus acciones en base a las preferencias del usuario.

6.4 Servidor central

El servidor central se encarga de centralizar, gestionar y garantizar la persistencia de datos relacionados a las puertas, los perfiles de usuario, los dispositivos móviles, y los dispositivos externos. Dentro de las responsabilidades de este componente se encuentra el despliegue y gestión de la red segura VPN, que funciona como el canal de comunicación principal entre las puertas y el servidor.

El servidor central también cumple el rol de funcionar como único enlace entre la puerta y los otros componentes de la arquitectura, extendiendo las funcionalidades de la puerta al permitirle interactuar con dispositivos externos diseñados para el internet de las cosas.

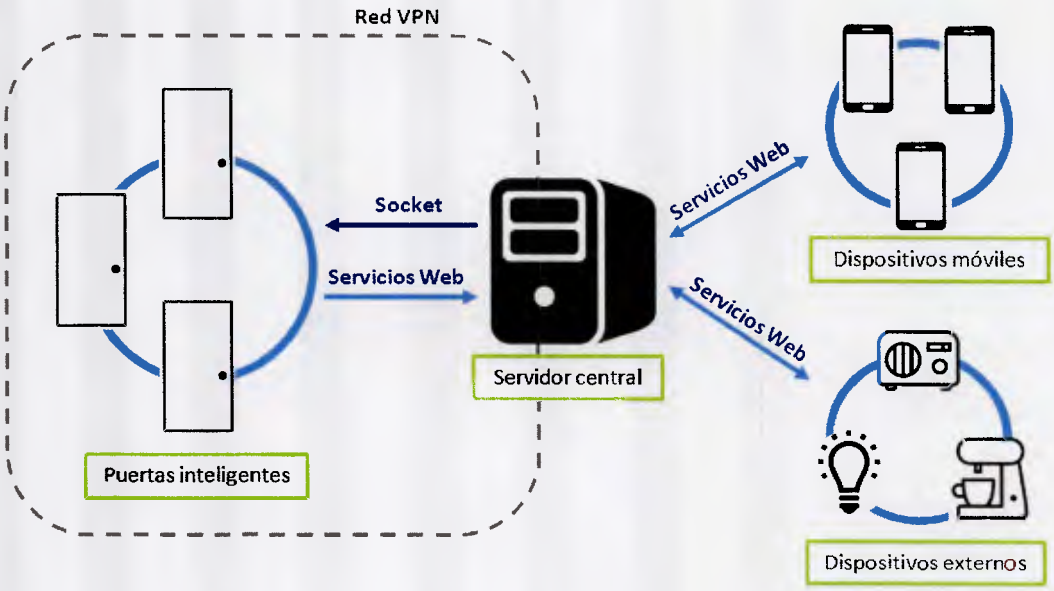


Figura 20: Métodos de comunicación de la solución
Fuente: Elaboración propia

6.6 Interacción con dispositivos del internet de las cosas

La solución está diseñada para soportar integraciones con dispositivos externos a la solución. De esta manera son habilitadas nuevas interacciones, servicios y funcionalidades para el usuario final.

Para demostrar las capacidades de integración de la solución, se diseñaron dos simuladores de dispositivo externo para el internet de las cosas, estos son:

- **Lámpara:** Se desarrolló una aplicación para el encendido remoto de una lámpara utilizando servicios web.

- **Reproductor de música:** Se desarrolló una aplicación que permite la reproducción de canciones de manera remota utilizando servicios web.

Se realizó la integración de estos dispositivos y se crearon nuevos escenarios de uso en base a la interacción de los mismos con la puerta. Cada usuario registrado en el prototipo de la puerta tiene la posibilidad de configurar distintas interacciones automáticas disparadas al realizar la apertura de la puerta. Las acciones automáticas disponibles son:

- Encendido de prototipo de lámpara construido.
- Reproducción de canción seleccionada en reproductor de música.
- Envío de correo electrónico con contenido y destinatario pre-configurados.

6.7 Cobertura de objetivos planteados

El presente trabajo de grado alcanzó los resultados esperados, cubriendo la totalidad de los objetivos y el alcance planteados. A continuación se describe brevemente la cobertura de estos.

- **Construir una puerta de seguridad, integrando electrónicamente los siguientes dispositivos: nodo de cómputo, cámara, lector de huellas dactilares, pestillo eléctrico y teclado numérico.**

  El prototipo de puerta construido cuenta con todos los dispositivos mencionados, y hace uso estos dispositivos para habilitar sus

características funcionales principales. Véase apartado 4.2 Iteración uno (1) - Construcción de prototipo de la puerta.

- **Desarrollar una aplicación para el nodo de cómputo que controle la apertura del pestillo, utilizando tecnologías de autenticación como reconocimiento facial, lectura de huellas dactilares y contraseña numérica.**

  Se desarrolló la aplicación para el nodo de cómputo, la cual cuenta con la capacidad de aperturar el pestillo eléctrico de forma automatizada, como también autenticar a los usuarios registrados en la aplicación haciendo uso de reconocimiento facial, lectura de huellas dactilares y contraseña numérica. Véase apartado 4.3 Iteración dos (2) - Desarrollo de software de la puerta.

- **Desarrollar una aplicación de servidor central que permita gestionar perfiles de usuario, preferencias y la comunicación e interacción entre la puerta y otros dispositivos vía internet.**

  Se desarrolló una aplicación de servidor central que permite la gestión de perfiles de usuario, al mismo tiempo que habilita la comunicación e interacción segura entre la puerta y dispositivos externos a la solución vía internet. Véase apartado 4.4 Iteración tres (3) - Desarrollo de servidor central.

- **Desarrollar una aplicación móvil que permita la configuración y apertura remota de la puerta.**

  Se desarrolló una aplicación móvil que permite la configuración de las preferencias de autenticación del usuario en la puerta, como también la gestión de la información básica del usuario. Entre las funcionalidades implementadas, se cuenta con la posibilidad de realizar la apertura remota de la puerta. Véase apartado 4.5 Iteración cuatro (4) - Desarrollo de aplicación móvil.

- **Evaluar servicios de mensajería en la capa de software entre la puerta y el servidor central en base a seguridad e implementar el mejor de ellos.**

  Se evaluaron diferentes tecnologías de comunicación entre la puerta y el servidor en base a seguridad. Se implementó una combinación de tecnologías como servicios web seguros, conexión por socket segura y red VPN. Véase apartado 4.3.1.1 Evaluación de tecnologías de comunicación.

- **Evaluar combinaciones de métodos de seguridad implementados (clave, reconocimiento facial y huellas dactilares) en base a seguridad y velocidad.**

  Una vez finalizado el proyecto se realizaron evaluaciones de rendimiento y seguridad para los métodos de autenticación implementados. Véase apartado 4.3.2.1 Evaluación de métodos de autenticación.

- **Construir un prototipo de la puerta a escala real.**

  Se construyó un prototipo de puerta inteligente a escala real. El mismo cuenta con un marco independiente que le permite abrir y cerrar sin necesidad de ser instalado en uno fijo. Véase apartado 4.2 Iteración uno (1) - Construcción de prototipo de la puerta.

- **Crear una prueba de concepto del proyecto en donde se demuestren dos escenarios de funcionalidad.**

  Se desarrollaron y construyeron los componentes necesarios para llevar a cabo tres escenarios de uso, superando el alcance definido, como prueba de concepto de la solución construida. Para mayor detalle sobre los escenarios ejecutados referirse al apartado 4.7 Pruebas de concepto.

Capítulo VII - Conclusiones

El internet de las cosas constituye unas de las tendencias más ambiciosas en los últimos tiempos, y consigo trae grandes oportunidades al mismo tiempo que retos a los encargados de hacerla una realidad.

Durante este trabajo de grado no solo se contemplaron los aspectos teóricos técnicos necesarios para diseñar una puerta inteligente para el internet de las cosas, sino que también se pasó a la práctica construyendo un prototipo y desarrollando todos los componentes de software necesarios para desplegar la solución, llevando a cabo una prueba de concepto diseñada en base a escenarios de uso cercanos a la realidad, facilitando el proceso de evaluación y mejora del producto resultante.

La expectativa del equipo es que los resultados alcanzados y descritos en este trabajo sirvan de insumo y base para esfuerzos similares en el futuro, brindando un panorama detallado sobre las decisiones de diseño de software, hardware, y arquitectura tomadas por el equipo, como también los componentes, herramientas, y plataformas utilizadas para desplegar la solución de puerta inteligente para el internet de las cosas.

7.1 Conclusiones

- Una solución para el internet de las cosas requiere un modelo lógico y de persistencia de datos flexible que permita realizar modificaciones recurrentes sin representar un alto impacto en la solución. Véase el apartado 5.1.1 Interoperabilidad.

- Una arquitectura centralizada facilita la integración de nuevos dispositivos al no requerir modificaciones ni actualizaciones en los dispositivos que se encuentran al borde de la arquitectura. Véase el apartado 5.1.1 Interoperabilidad.

Las soluciones para el internet de las cosas manejan un alto volumen de datos de datos y transacciones por lo que es importante tomar en cuenta la escalabilidad de la arquitectura diseñada. Véase el apartado 5.1.2 Escalabilidad.Esta integración con dispositivos externos también expande la arquitectura inicial de la solución, en el diagrama de despliegue a continuación se puede observar el impacto de dicha integración en la arquitectura original.

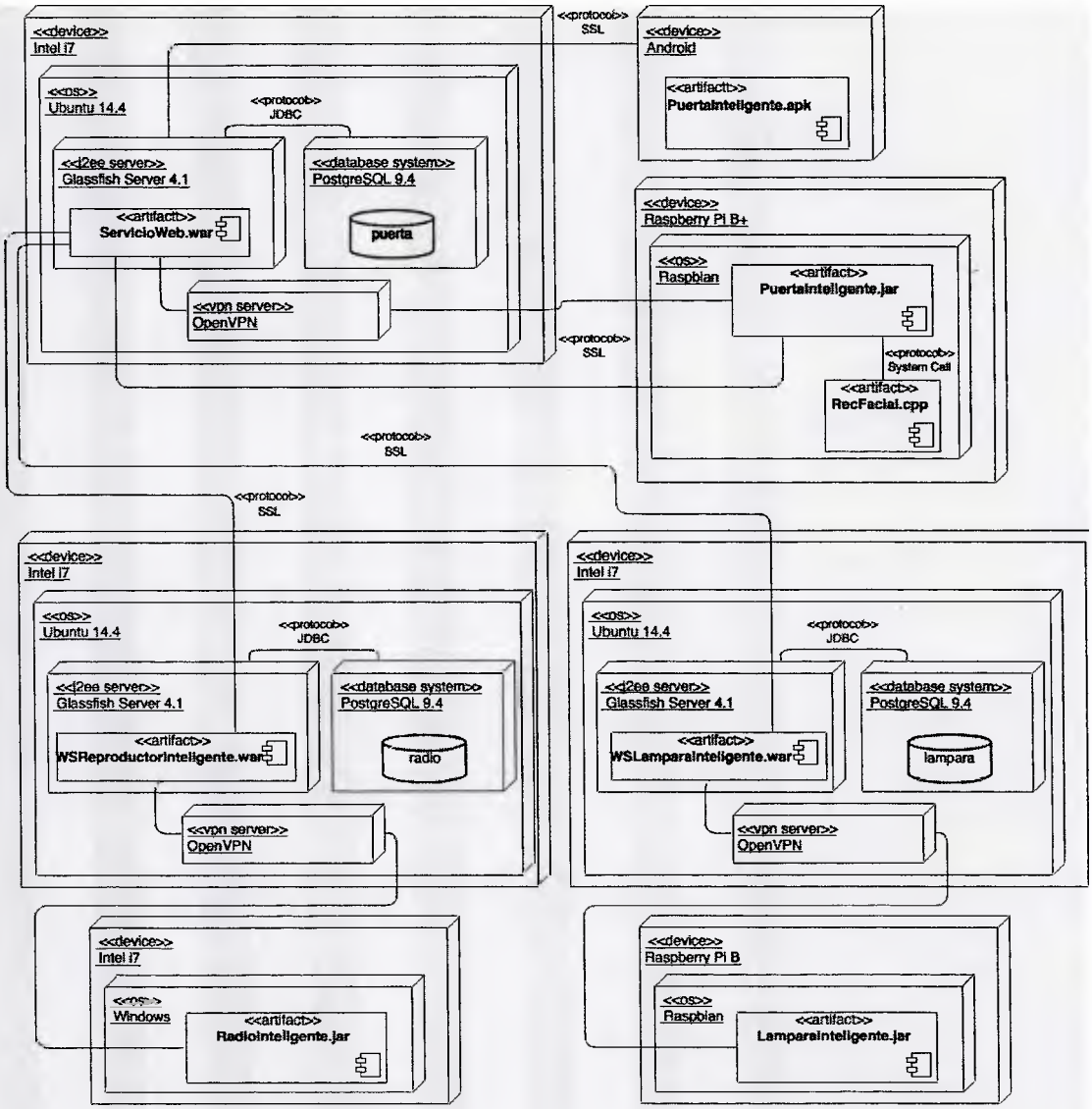


Figura 14: Arquitectura de la solución con dispositivos externos
Fuente: Elaboración propia

- 5.1.2 Escalabilidad

- La construcción de prototipos y las pruebas de concepto brindan una gran cantidad de ventajas a la hora de evaluar la factibilidad y experiencia de uso general de un concepto de solución. Véase el apartado 4.2 Iteración uno (1) - Construcción de prototipo de la puerta.

- Es factible desarrollar soluciones para el internet de las cosas utilizando las tecnologías existentes, aunque sigue presente la necesidad de definir un estándar de comunicación entre este tipo de soluciones para facilitar el esfuerzo de integración. Véase apartado 5.1.1 Interoperabilidad.

- Debido a la naturaleza distribuida de este tipo de soluciones, debe hacerse énfasis en lograr un alto grado de seguridad y encapsulamiento en cada uno de los componentes de la solución. Véase apartado 5.5 Seguridad y privacidad.

- Las conexiones por sockets de internet brindan una alternativa de bajo consumo de recursos, y alto grado de interoperabilidad y seguridad para conexiones de punto a punto que no requieran manejar alto volumen de peticiones. Véase apartado 4.3.1.1 Evaluación de tecnologías de comunicación

- Los servicios web brindan un alto grado de interoperabilidad y un buen nivel de eficiencia al manejar un alto volumen de peticiones representando una alternativa ideal para servidores centralizados que

requieran de escalabilidad. Véase apartado 4.3.1.1 Evaluación de tecnologías de comunicación

- La mejor combinación en base a seguridad de 2 métodos de autenticación, considerando únicamente los métodos evaluados en este proyecto, es el uso de autenticación por clave numérica y lectura de huella dactilar. Véase apartado 4.3.2.1 Evaluación de métodos de autenticación.

7.2 Recomendaciones

Si bien se alcanzaron todos los objetivos definidos para este trabajo de grado, existen una gran cantidad de posibilidades de investigación y mejoras para los resultados alcanzados. A continuación se describen algunas de las recomendaciones planteadas por el equipo, en base a mejoras a la solución actual y en nuevas posibilidades y aplicaciones:

- Creación de solución de gestión web con interfaz gráfica para administrar la información de usuarios almacenada en el servidor central.
- Creación de reportes gráficos que presenten información sobre interacciones, usuarios, y uso de las puertas, tanto para el usuario final como para la gestión total del sistema.
- Despliegue del servidor central bajo un modelo de computación en la nube, y ejecución de pruebas de estrés para medir las capacidades de escalamiento del mismo.
- Aplicación de tecnologías de Big Data para el almacenamiento, descubrimiento y análisis de información no estructurada generada por la solución.
- Actualización del hardware utilizado para el nodo de cómputo.
- Implementación de métodos en el nodo de cómputo para conexión automática a red WiFi del usuario como primer paso de instalación del dispositivo.

- Uso del lector de huellas dactilares presente en nuevos dispositivos móviles para realizar la autenticación de huella dactilar de manera remota.
- Uso de la cámara presente en nuevos dispositivos móviles para realizar la autenticación facial de manera remota.
- Aplicación de algoritmos de inteligencia artificial para la detección de manera proactiva de posibles ataques y vulnerabilidades en la red.
- Implementación de tecnologías de autenticación adicionales en la puerta como: RFID, NFC, reconocimiento de voz, entre otras.
- Uso de batería de alta carga para garantizar autonomía de la puerta en caso de pérdida de corriente por tierra.
- Uso de cámara de alta definición para mejorar precisión en reconocimiento facial.
- Implementación de base de datos NoSQL Berkley DB para dispositivos en el borde de la arquitectura.
- Utilizar cerraduras inalámbricas que hagan uso de tecnologías y protocolos de comunicación de bajo consumo como ZigBee, Z-wave, entre otros.
- Implementar un patrón de mensajería de publicación-suscripción utilizando tecnologías como Paho o MQTT, el mismo representa múltiples beneficios para comunicaciones M2M (máquina a máquina por sus siglas en inglés).

Referencias

[1] Hay, M. (12 de 8 de 2014). 3 Disruptive Trends In 2015: IoT Leads List. Obtenido de informationweek.com: http://www.informationweek.com/big-data/big-data-analytics/3-disruptive-trends-in-2015-iot-leads-list/a/d-id/1317936

[2] Gartner. (s.f.). The Internet of Things - Internet of Things Companies. Obtenido de gartner.com: http://www.gartner.com/it-glossary/internet-of-things/

[3] PHILIPS. (s.f.). Meet hue. Obtenido de meethue.com: http://www2.meethue.com/en-us/

[4] nest. (s.f.). Home Nest. Obtenido de nest.com: https://nest.com/

[5] Enevo Inc. (s.f.). Enevo. Obtenido de Enevo Waste Collection for Smart Cities: http://www.enevo.com/

[6] GREENOUGH, J. (s.f.). How The Internet Of Things Market Will Grow. Obtenido de Business Insider: http://www.businessinsider.com/how-the-internet-of-things-market-will-grow-2014-10

[7] Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges. (2012). Obtenido de computer.org: http://www.computer.org/web/csdl/index/-/csdl/proceedings/fit/2012/4946/00/4927a257.pdf

[8] Ashton, K. (22 de 06 de 2009). That 'Internet of Things' Thing. Obtenido de rfidjournal.com: http://www.rfidjournal.com/articles/view?4986

[9] Gershenfeld, N. (s.f.). Neil Gershenfeld. Obtenido de Wikipedia: http://en.wikipedia.org/wiki/Neil_Gershenfeld

[10] Postscapes. (s.f.). History of the Internet of Things. Obtenido de Postscapes: http://postscapes.com/internet-of-things-history

[11] International Telecommunication Union. (11 de 2005). The Internet of Things. Obtenido de International Telecommunication Union: http://www.itu.int/osg/spu/publications/internetofthings/InternetofThings_summary.pdf

[12] National Intelligence Council. (04 de 2008). Disruptive Civil Technologies. Obtenido de National Intelligence Council: http://fas.org/irp/nic/disruptive.pdf

[13] Evans, D. (04 de 2011). The Internet of Things How the Next Evolution of the Internet Is Changing Everything. Obtenido de Cisco: http://www.cisco.com/web/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

[14] Guthikonda, R. T., Chitta, S. S., Tekawade, S., & Attavar, T. (25 de 04 de 2014). COMPARATIVE ANALYSIS OF IoT ARCHITECTURES. Obtenido de beboulderanywhere: http://beboulderanywhere.colorado.edu/itp/sites/default/files/attachments/itp/team_8_-_shraddha_subhash_tekawade_-_apr_25_2014_453_pm_-_group8_final_paper.pdf

[15] Agile Modeling. (s.f.). Feature Driven Development (FDD) and Agile Modeling. Obtenido de Agile Modeling: http://www.agilemodeling.com/essays/fdd.htm

[16] Palmer, S. R. (2014). FDD History. Obtenido de step-10: http://www.step-10.com/SoftwareProcess/FeatureDrivenDevelopment/FDDHistory.html

[17] HERNANDEZ, D. R. (10 de 07 de 2006). La estatura no es problema en Venezuela. Obtenido de EL UNIVERSAL: http://www.eluniversal.com/2006/07/10/ten_art_09454A

[18] Juric, M. B., Rozman, I., Brumen, B., Colnaric, M., & Hericko, M. (16 de 09 de 2005). Comparison of performance of Web services, WS-Security, RMI, and RMI–SSL. Obtenido de semgrid: http://www.semgrid.net/Citation-Before-2006.1/+++JSS-2006-Service.pdf

[19] Kale, H., & Patel, J. (2012). Performance Comparison Of Java Remote Method Invocation VS Java Messaging Service. Obtenido de RTI: http://www.cs.rit.edu/~hsk5260/Final%20Report.pdf

[20] OpenCV. (s.f.). Face Recognition with OpenCV. Obtenido de OpenCV: http://docs.opencv.org/modules/contrib/doc/facerec/facerec_tutorial.html

[21] Digital Persona. (s.f). DigitalPersona White Paper Guide to Fingerprint Recognition. Obtenido de https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Studies/BioFinger/BioFinger_pdf.pdf?__blob=publicationFile

Anexos

Tabla de anexos

Anexo A - Instrumentos FDD 1

A.1 Lista de actividades por área 1

A.2 Lista de funcionalidades 2

A.3 Roadmap del proyecto 4

Anexo B - Modelos de dominio 5

B.1 Modelo de dominio puerta 5

B.2 Modelo de dominio servidor central 6

B.3 Modelo de dominio aplicación móvil 7

Anexo C - Diagramas de clases 8

C.1 Diagrama de clases puerta 8

C.2 Diagrama de clases servidor 9

C.3 Diagrama de clases aplicación móvil 10

Anexo D - Modelo de base de datos 11

D.1 Modelo de base de datos relacional para servidor central 11

Anexo E - Diagramas de arquitectura 12

E.1 Diagrama de componentes 12

E.2 Diagrama de despliegue 13

Anexo F - Maquetas para aplicación móvil 14

Anexo G - Diagramas de secuencia 16

G.1 Diagrama de secuencia proceso de registro 16

G.2 Diagrama de secuencia apertura remota 17

Anexo H - Fotos detalladas del prototipo puerta 18

Anexo I - Componentes utilizados en prototipo 20

I.1 Lista detallada de componentes utilizados 20
I.2 Diseño de circuitos electrónicos 22
I.2.1 Circuito de apertura del pestillo eléctrico 22
I.2.2 Circuito para sensor de efecto Hall 23
I.2.3 Circuito de LED para iluminación en la puerta 23
I.3 Diagrama de cableado y distribución de componentes 24
Anexo J - Evaluación de cerraduras de seguridad 25

Índice de figuras

Figura 1: Roadmap del proyecto 4
Figura 2: Modelo de dominio puerta 5
Figura 3: Modelo de dominio servidor central 6
Figura 4: Modelo de dominio aplicación móvil 7
Figura 5: Diagrama de clases puerta 8
Figura 6: Diagrama de clases servidor 9
Figura 7: Diagrama de clases aplicación móvil 10
Figura 8: Modelo de base de datos relacional para servidor central 11
Figura 9: Diagrama de componentes 12
Figura 10: Diagrama de despliegue 13
Figura 12: Maqueta y resultado pantalla de login 14
Figura 11: Maqueta y resultado pantalla de inicio 14
Figura 15: Maqueta pantalla de configuración 15
Figura 14: Maqueta pantalla de vista remota 15
Figura 13: Maqueta y resultado pantalla de menú 15
Figura 16: Diagrama de secuencia proceso de registro 16
Figura 17: Diagrama de secuencia apertura remota 17
Figura 18: Vista frontal del prototipo de la puerta 18
Figura 19: Vista trasera del prototipo de la puerta 18

Figura 21: Zona de procesamiento y componentes eléctricos 19
Figura 20: Sensor magnético y pestillo eléctrico 19
Figura 22: Circuito de apertura del pestillo eléctrico 22
Figura 23: Circuito para sensor de efecto Hall 23
Figura 24: Circuito de LED para iluminación en la puerta 23
Figura 25: Diagrama de cableado y distribución de componentes 24
Figura 26: Cerradura electromagnética 25
Figura 27: Abrepuertas eléctrico Fuente: andreu.es 27
Figura 28: Pestillo eléctrico 27
Figura 29: Cerradura electromecánica 28

Índice de tablas

Tabla 1: Lista de actividades por área FDD 1
Tabla 2: Lista de funcionalidades FDD. Elaboración propia 3

Anexo A - Instrumentos FDD

A.1 Lista de actividades por área

Lista de actividades por área - FDD

Áreas							
P	Puerta	S	Servidor	M	Aplicación Móvil	D	Dispositivos Externos
ID	Actividades	ID	Actividades	ID	Actividades	ID	Actividades
1	Construcción de prototipo puerta	1	Comunicación con puertas	1	Consulta de estado	1	Construcción de prototipo lámpara
2	Clave numérica	2	Comunicación con dispositivos móviles	2	Control remoto de la puerta	2	Encendido de lámpara remoto
3	Lectura de huellas	3	Integración con dispositivos externos	3	Botón de pánico	3	Reproducción de MP3 remoto
4	Reconocimiento Facial	4	Servidor de aplicaciones	4	Historial de eventos	4	Envío de correos en base a eventos
5	Autenticación de usuarios	5	Persistencia	5	Gestión de usuarios		
6	Apertura de pestillo	6	Red VPN	6	Gestión de dispositivos		
7	Timbre inteligente	7	Seguridad central y enrutamiento	7	Configuración de puerta		
8	Sonidos de respuesta						
9	Gestión de usuarios						
10	Configuración						
11	Persistencia de datos						
12	Comunicación externa						

Tabla 1: Lista de actividades por área FDD
Fuente: Elaboración propia

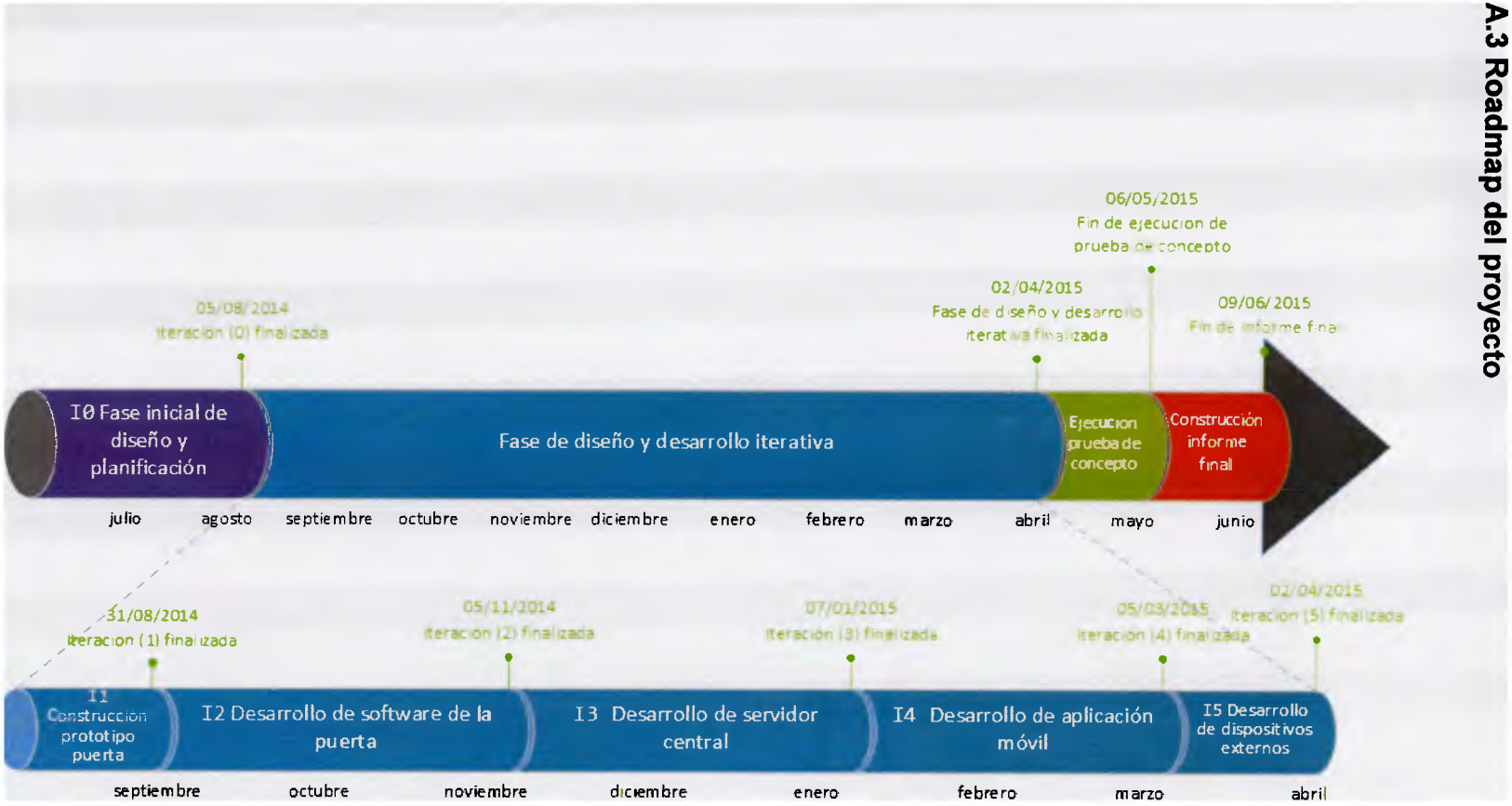
A.2 Lista de funcionalidades

Lista de funcionalidades - FDD		
ID	Área	Funcionalidad
D1-1	Dispositivos externos	Ensamblar componentes
D2-1	Dispositivos externos	Encender lámpara de manera automatizada
D2-2	Dispositivos externos	Exponer servicio para encender lámpara de manera remota
D3-1	Dispositivos externos	Reproducir MP3 de manera automatizada
D3-2	Dispositivos externos	Exponer servicio para reproducción de MP3 de manera remota
D4-1	Dispositivos externos	Enviar correos de manera automatizada
M1-1	Aplicación móvil	Solicitar estado de la puerta
M2-1	Aplicación móvil	Ver vista remota de una puerta
M2-2	Aplicación móvil	Permitir al usuario abrir la puerta de manera remota
M3-1	Aplicación móvil	Permitir al usuario disparar proceso de pánico
M4-1	Aplicación móvil	Permitir al usuario ver historial de eventos recientes en la puerta
M5-1	Aplicación móvil	Ingresar a la aplicación móvil un usuario
M5-2	Aplicación móvil	Registrar en la aplicación móvil un usuario
M5-3	Aplicación móvil	Editar información básica de cuenta de usuario
M5-4	Aplicación móvil	Eliminar usuario de aplicación y puerta
M6-1	Aplicación móvil	Gestionar dispositivos conectados a la puerta
M7-1	Aplicación móvil	Validar con contraseña de la puerta funcionalidad
M7-2	Aplicación móvil	Configurar correos electrónicos de usuarios para funcionalidad
M7-3	Aplicación móvil	Configurar métodos de acceso activos en puerta
P1-1	Puerta	Listar componentes
P1-2	Puerta	Diseñar distribución de componentes
P1-3	Puerta	Diseñar circuitos
P1-4	Puerta	Construcción de circuitos
P1-5	Puerta	Ensamblado de componentes
P2-1	Puerta	Validar mediante una clave numérica la identidad de un usuario
P3-1	Puerta	Validar mediante detección de huella dactilar la identidad de un usuario
P4-1	Puerta	Validar mediante detección facial la identidad de un usuario
P4-2	Puerta	Encender una luz para mejorar iluminación en

		reconocimiento facial
P4-3	Puerta	Enviar foto de cámara integrada en la puerta al servidor
P5-1	Puerta	Integrar los distintos métodos de autenticación para abrir el pestillo de la puerta
P5-2	Puerta	Implementar menú de usuario utilizando sonidos y teclado
P6-1	Puerta	Abrir pestillo de la puerta
P7-1	Puerta	Disparar proceso de timbre inteligente
P7-2	Puerta	Emitir sonido de timbre
P8-1	Puerta	Emitir sonido de alarma
P8-2	Puerta	Disparar proceso de alarma
P9-1	Puerta	Gestión de usuarios
P10-1	Puerta	Configurar métodos de autenticación activos
P10-2	Puerta	Creación de script de arranque de la puerta
P11-1	Puerta	Persistencia de la información de configuración
P11-2	Puerta	Registro de eventos de la puerta
P11-3	Puerta	Persistencia de registro de eventos en puerta
P12-1	Puerta	Conexión por socket al servidor central
P12-2	Puerta	Conexión a VPN del servidor central
P12-3	Puerta	Envío recurrente de respaldo de eventos a servidor
S1-1	Servidor	Exposición de servicios web para puerta
S1-2	Servidor	Envió de peticiones por socket a puerta
S2-1	Servidor	Exposición de servicios web para aplicación móvil
S3-1	Servidor	Integración con lámpara
S3-2	Servidor	Integración con reproductor de música
S4-1	Servidor	Exposición en red pública de servicios
S4-2	Servidor	Habilitar únicamente conexión SSL a servicios
S4-3	Servidor	Despliegue de aplicación de servidor central
S5-1	Servidor	Persistencia de datos en BD
S6-1	Servidor	Inicializar red segura VPN
S6-2	Servidor	Recepción de conexiones por red VPN
S6-3	Servidor	Asignación de IPs a puertas
S7-1	Servidor	Realizar validaciones de control de acceso
S7-2	Servidor	Enrutamiento de peticiones

Tabla 2: Lista de funcionalidades FDD. Elaboración **propia**

A.3 Roadmap del proyecto



B.1 Modelo de dominio puerta

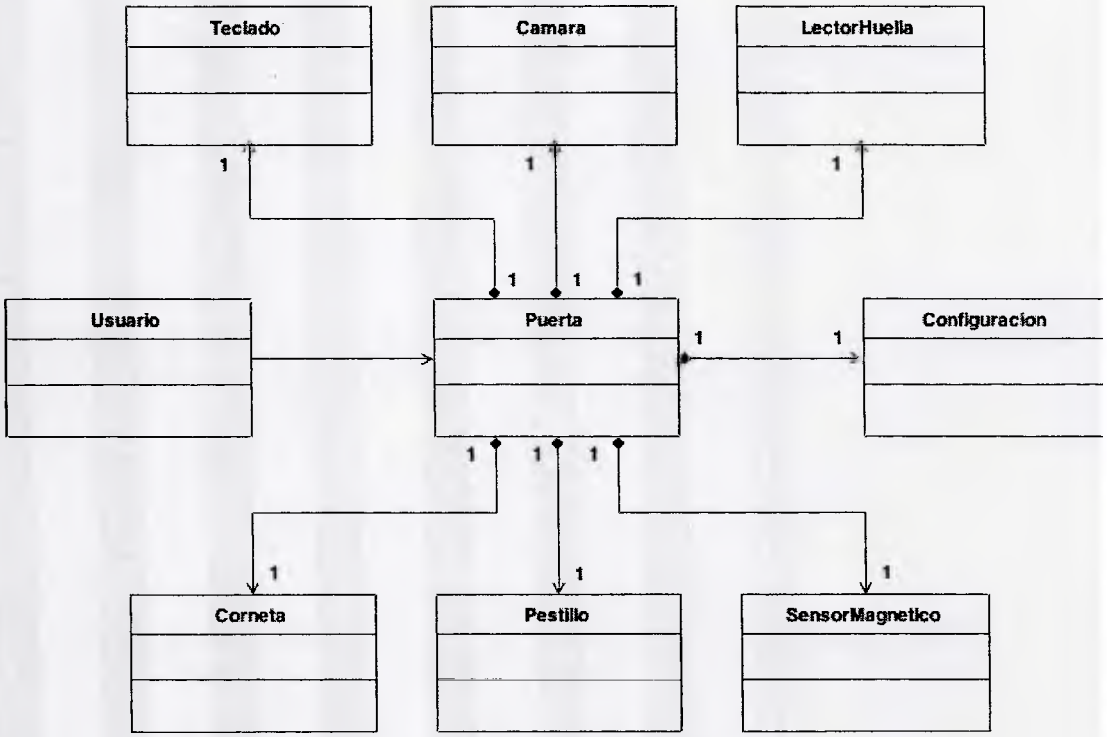


Figura 2: Modelo de dominio puerta
Fuente: Elaboración propia

B.2 Modelo de dominio servidor central

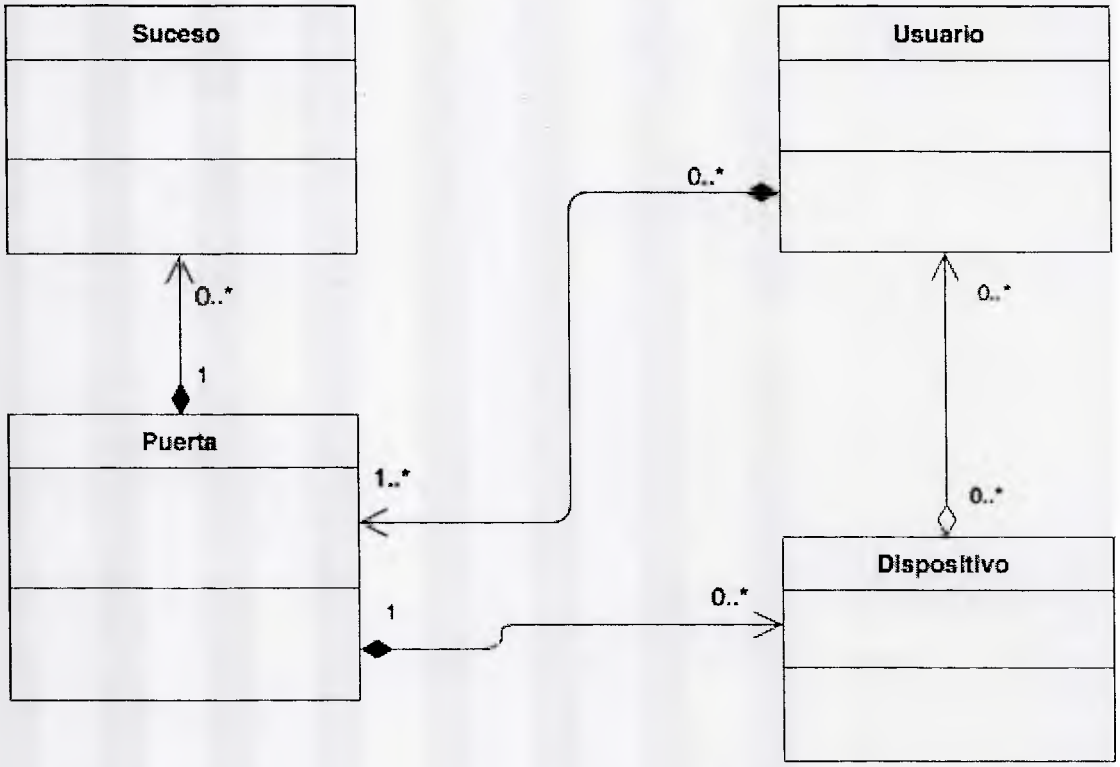


Figura 3: Modelo de dominio servidor central
Fuente: Elaboración propia

B.3 Modelo de dominio aplicación móvil

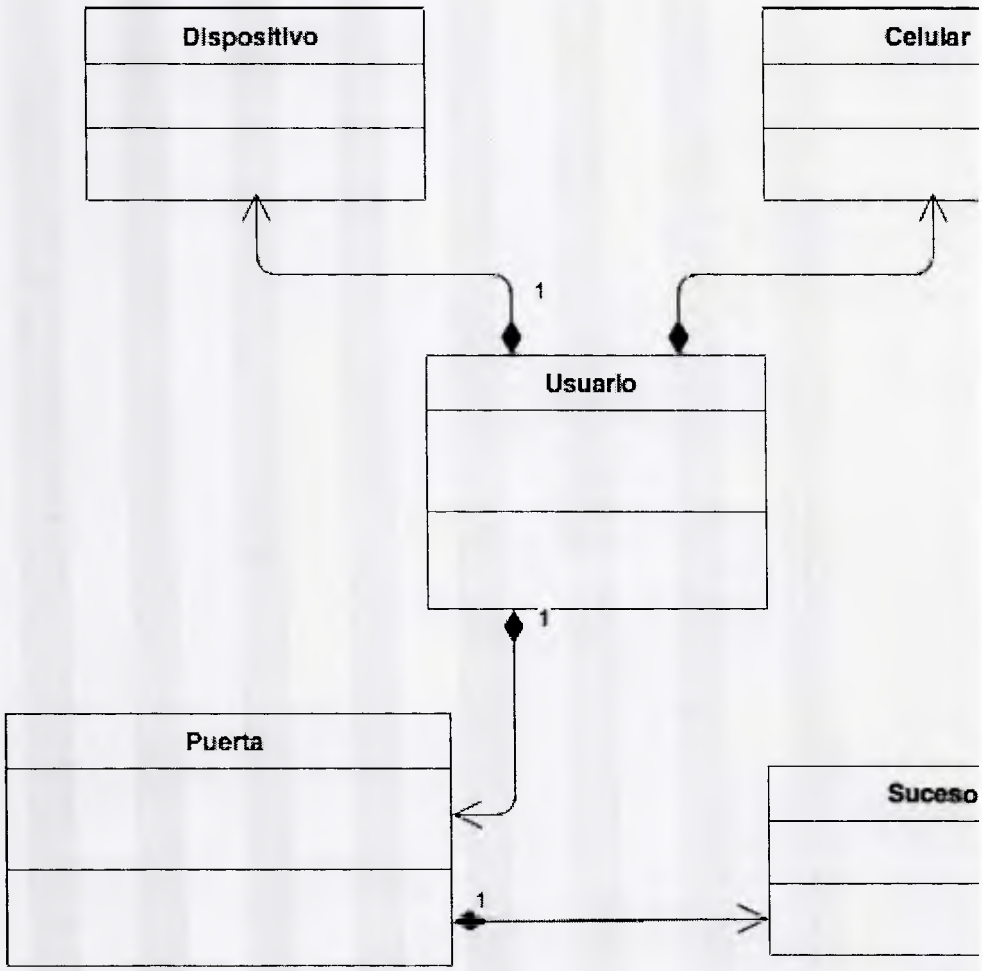


Figura 4: Modelo de dominio aplicación móvil
Fuente: Elaboración propia

C.1 Diagrama de clases puerta

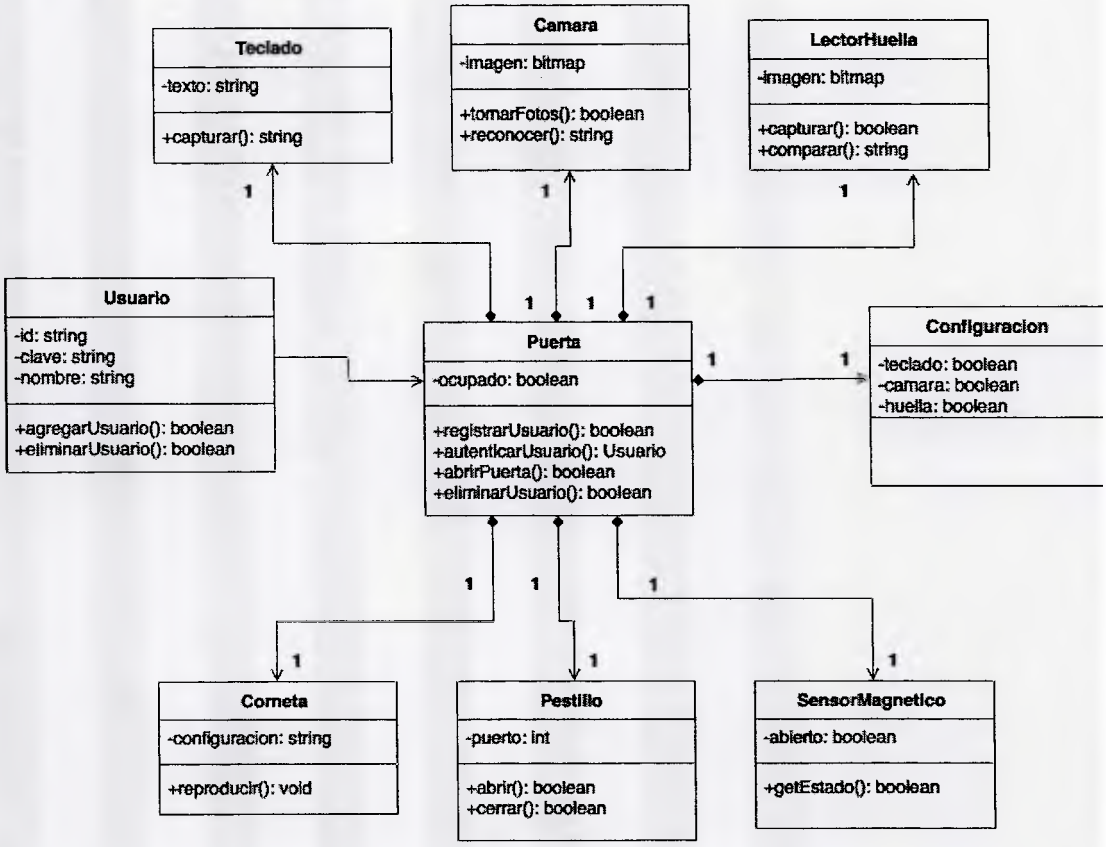


Figura 5: Diagrama de clases puerta
Fuente: Elaboración propia

C.2 Diagrama de clases servidor

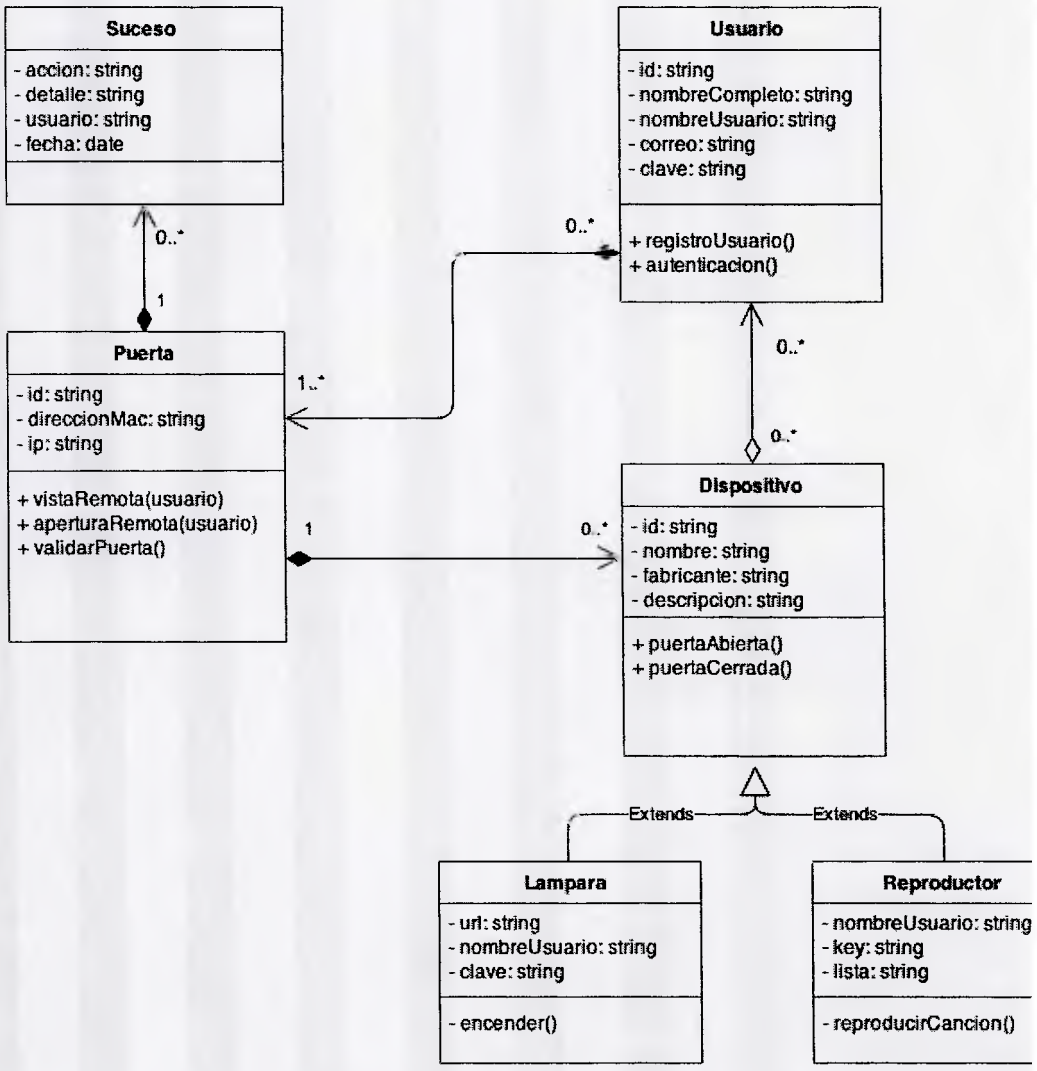


Figura 6: Diagrama de clases servidor
Fuente: Elaboración propia

C.3 Diagrama de clases aplicación móvil

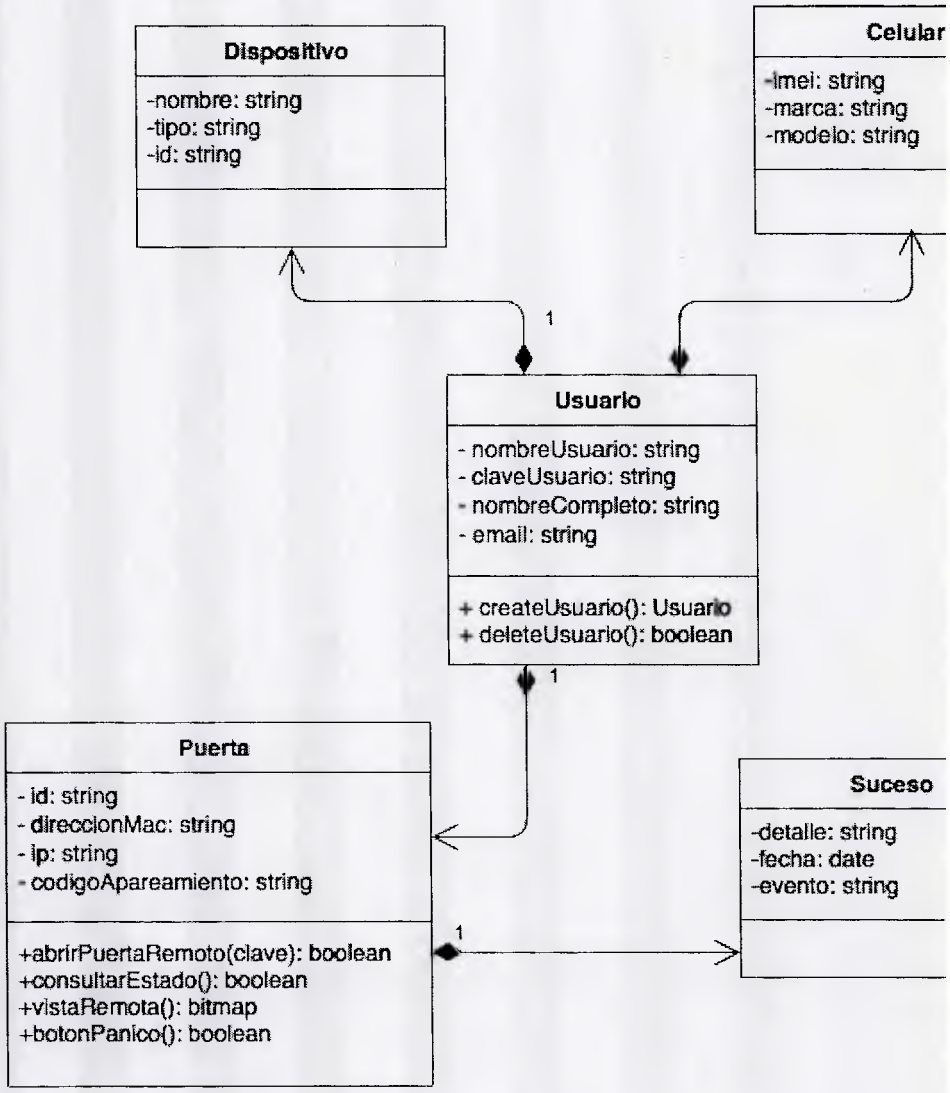


Figura 7: Diagrama de clases aplicación móvil
Fuente: Elaboración propia

Anexo D - Modelo de base de datos

D.1 Modelo de base de datos relacional para servidor central

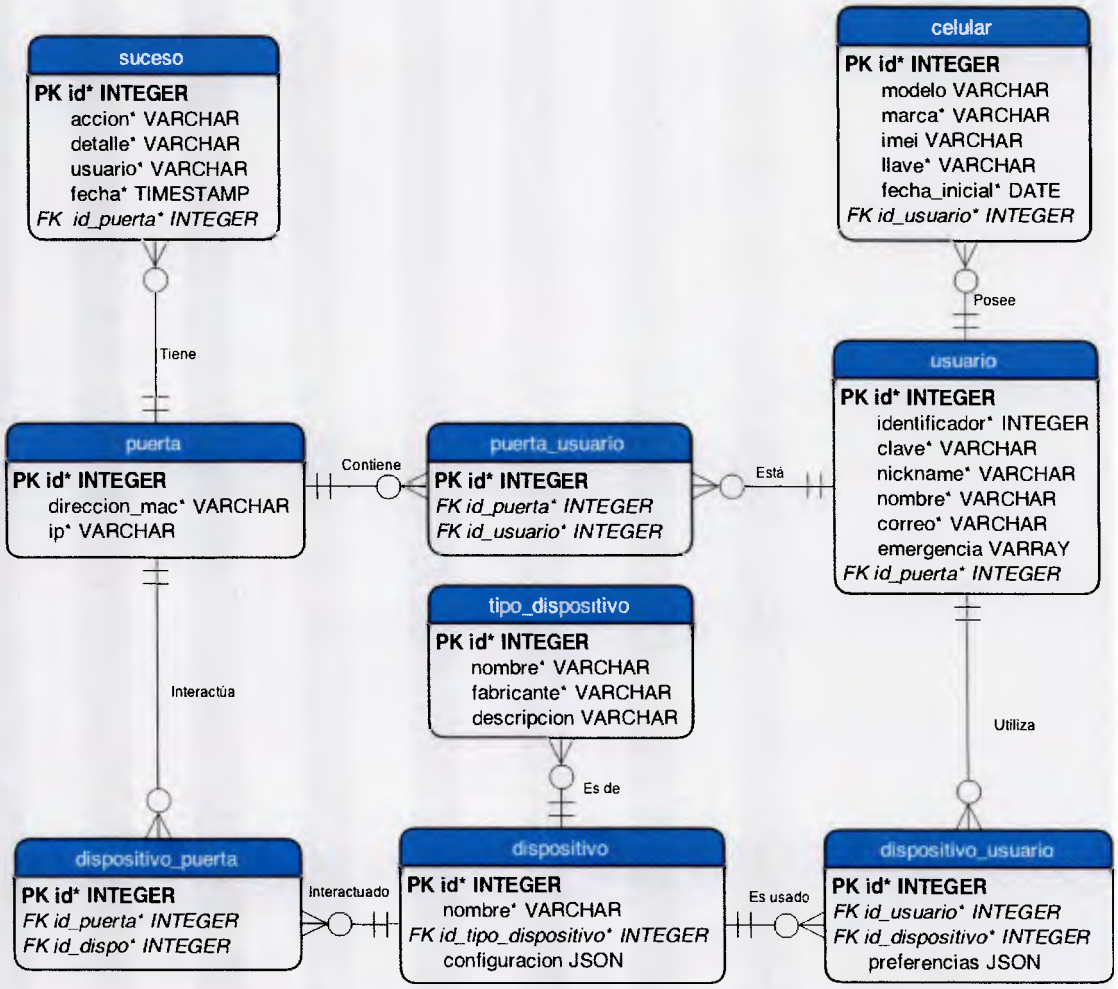


Figura 8: Modelo de base de datos relacional para servidor central
Fuente: Elaboración propia

Anexo E - Diagramas de arquitectura

E.1 Diagrama de componentes

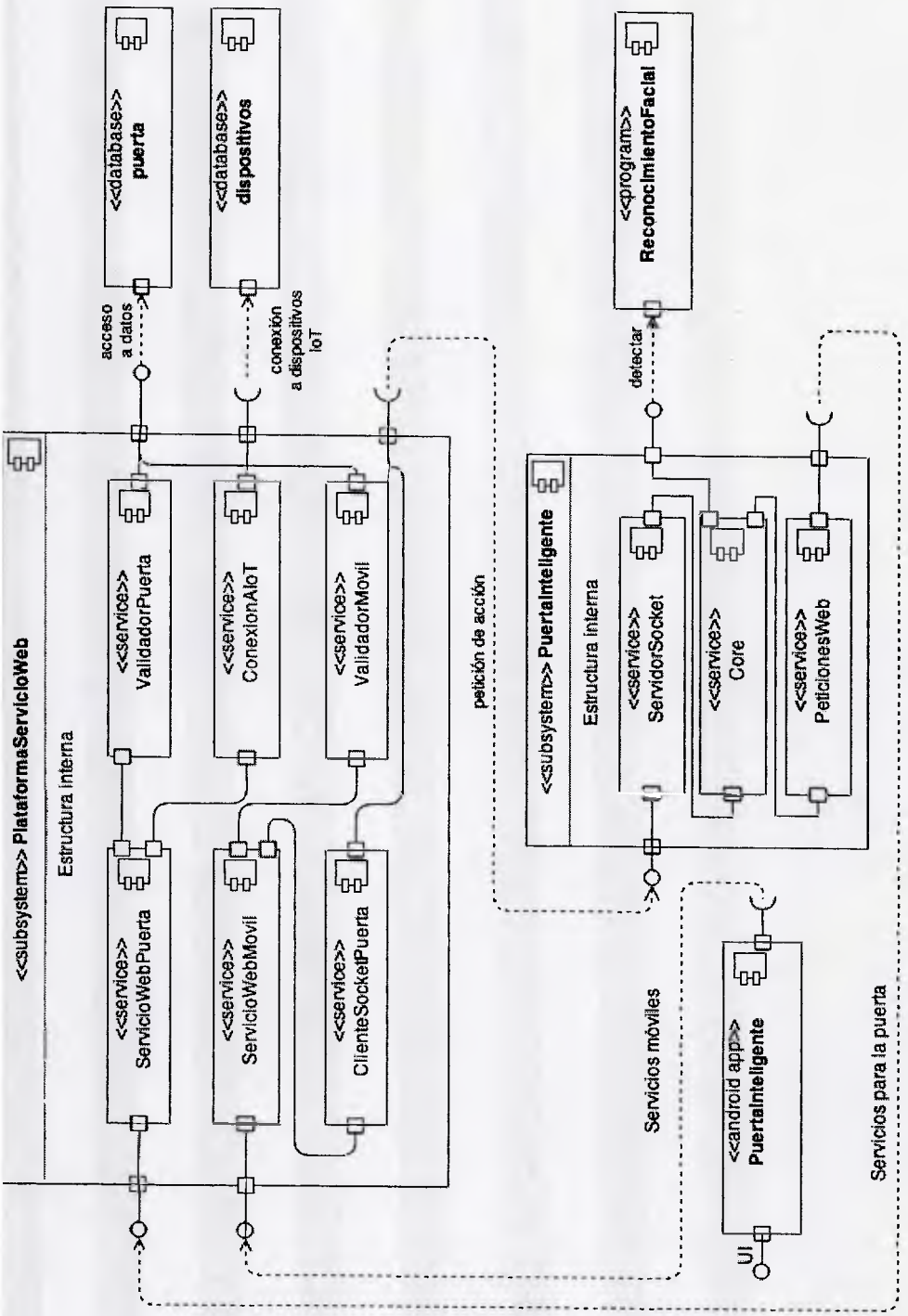


Figura 9: Diagrama de componentes
Fuente: Elaboración propia

E.2 Diagrama de despliegue

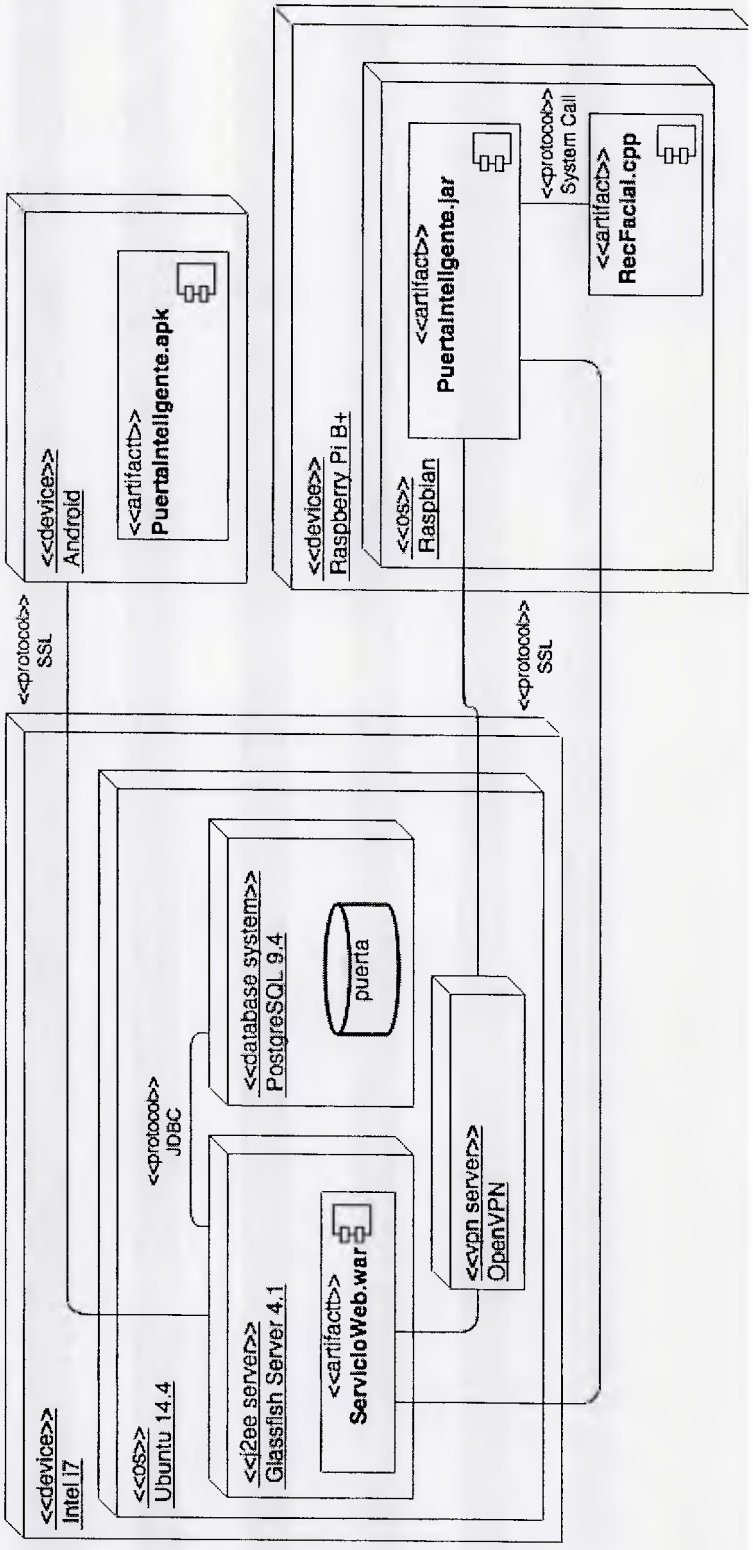


Figura 10: Diagrama de despliegue
Fuente: Elaboración propia

Anexo F - Maquetas para aplicación móvil

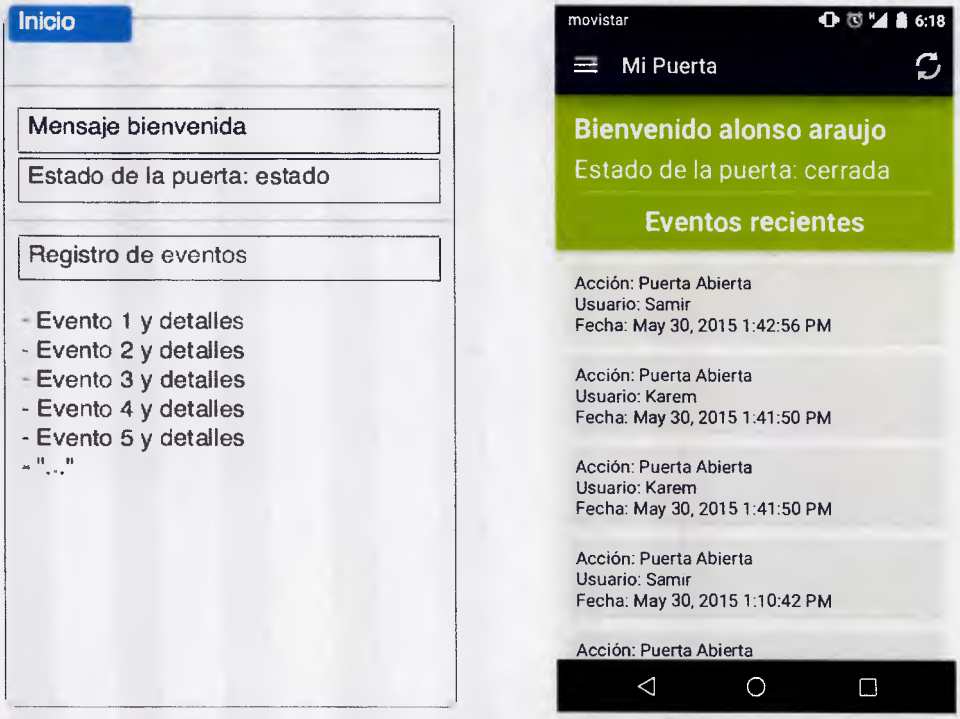


Figura 12: Maqueta y resultado pantalla de inicio
Fuente: Elaboración propia

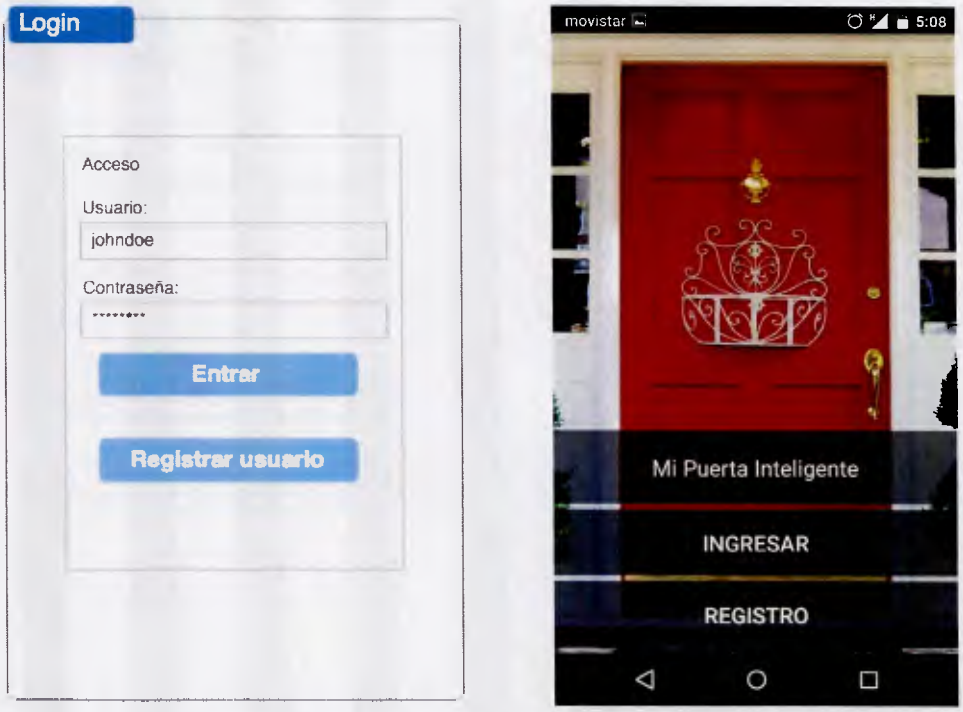


Figura 11: Maqueta y resultado pantalla de login
Fuente: Elaboración propia

Figura 15: Maqueta y resultado pantalla de menú
Fuente: Elaboración propia

Figura 14: Maqueta pantalla de vista remota
Fuente: Elaboración propia

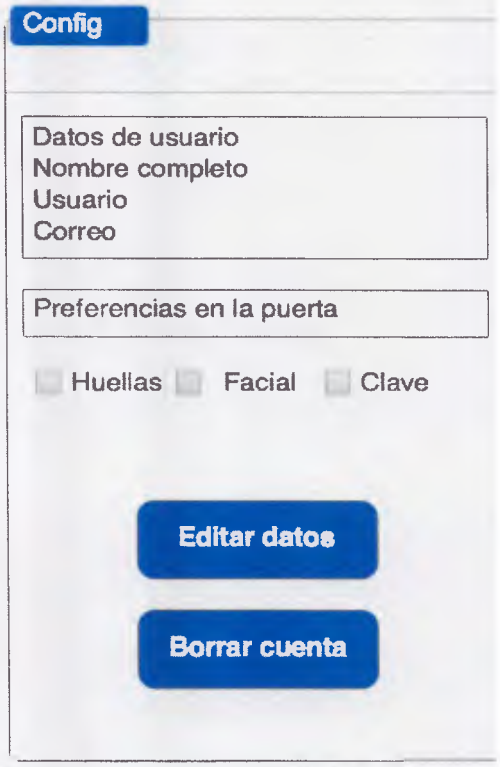


Figura 13: Maqueta pantalla de configuración
Fuente: Elaboración propia

G.1 Diagrama de secuencia proceso de registro

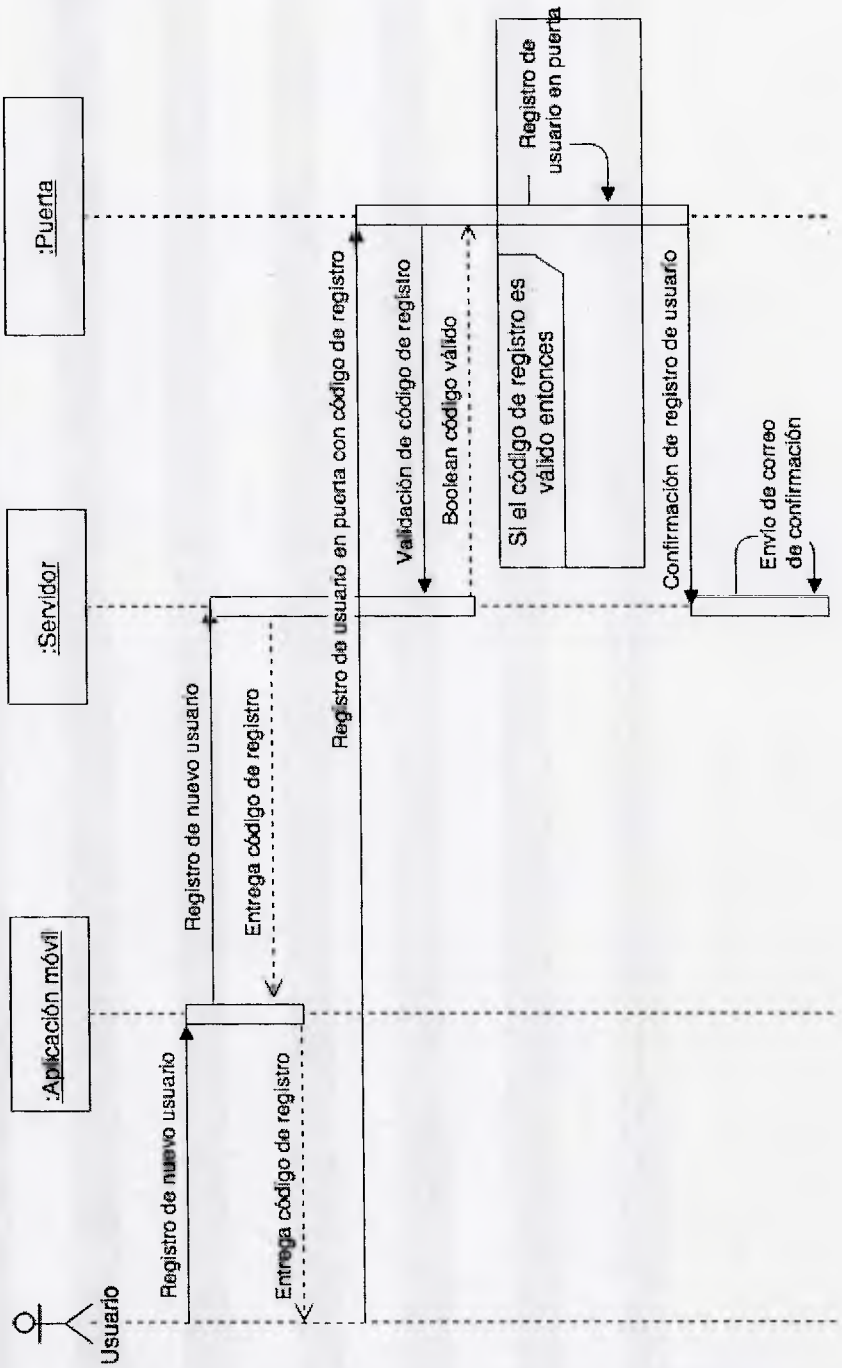


Figura 16: Diagrama de secuencia proceso de registro
Fuente: Elaboración propia

G.2 Diagrama de secuencia apertura remota

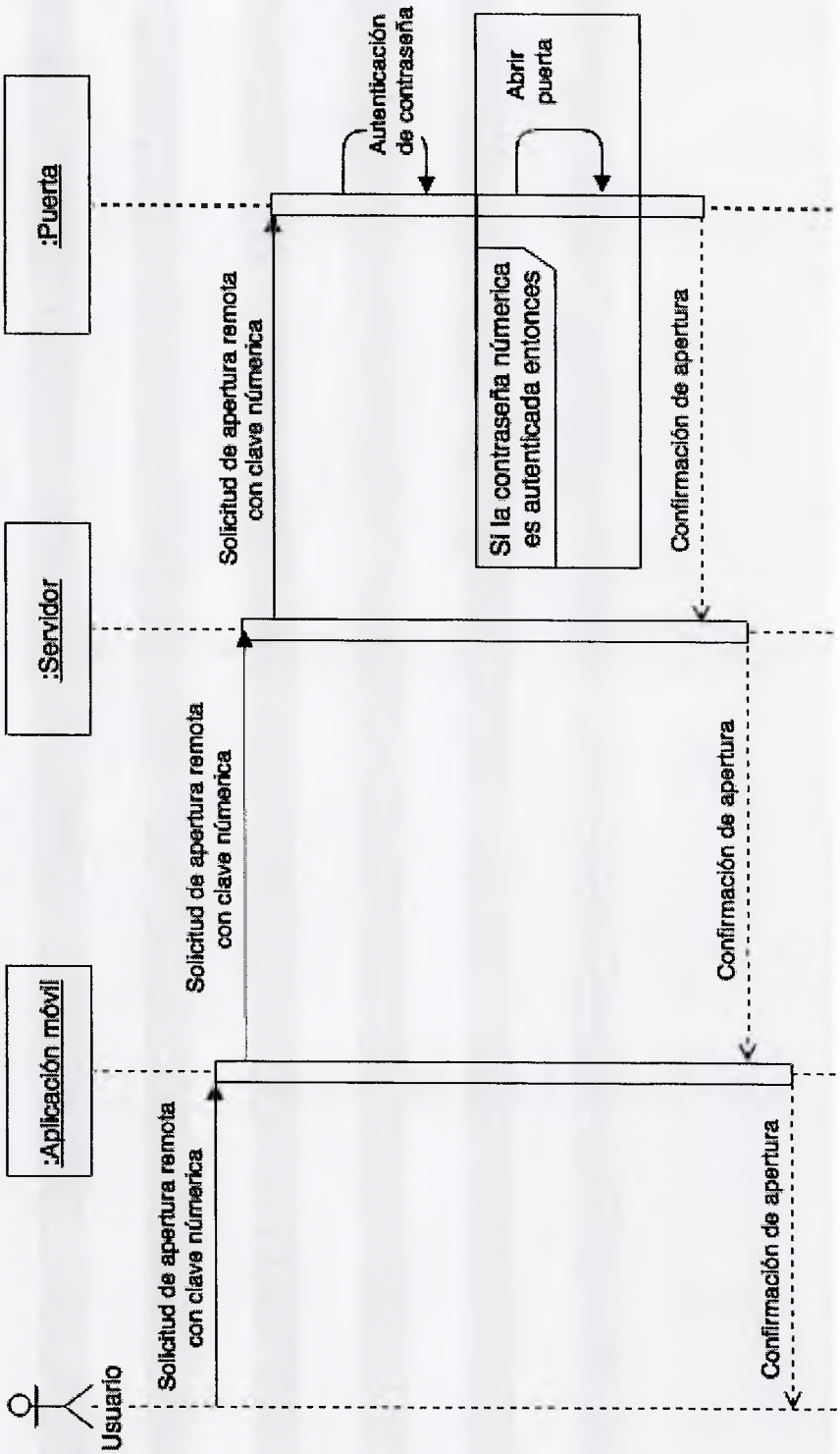


Figura 17: Diagrama de secuencia apertura remota
Fuente: Elaboración propia

Anexo H - Fotos detalladas del prototipo puerta

Figura 18: Vista frontal del prototipo de la puerta
Fuente: Elaboración propia

Figura 19: Vista trasera del prototipo de la puerta
Fuente: Elaboración propia

Figura 21: Sensor magnético y pestillo eléctrico
Fuente: Elaboración propia

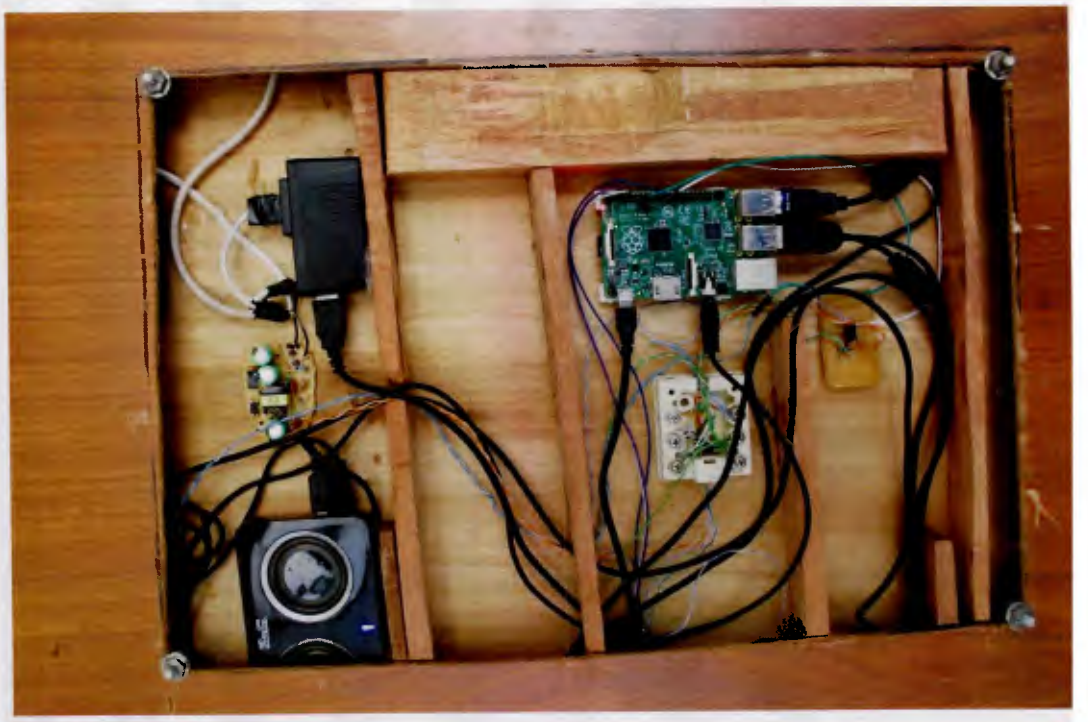


Figura 20: Zona de procesamiento y componentes eléctricos
Fuente: Elaboración propia

Anexo I - Componentes utilizados en prototipo

I.1 Lista detallada de componentes utilizados

- **Computadora:** se decidió el uso de una computadora Raspberry Pi B+ como nodo de cómputo para la puerta. Esta computadora es una de las más económicas del mercado, costando alrededor de 30$ para la fecha. Esta cuenta con cuatro salidas a puertos USB, un bus de GPIO, un procesador ARM de 700MHz, 512MB de memoria RAM y un bajo consumo de corriente (5V) que puede ser respaldado con una batería en caso de cortes de la fuente eléctrica principal sin ver afectado su rendimiento.
- **Cámara:** dispositivo que permite visualizar quién está del lado externo de la puerta. Se utilizo una Microsoft Lifecam vx-100.
- **Detector de huellas dactilares:** se decidió usar el Microsoft Fingerprint Reader ya que posee un hardware adecuado para el reconocimiento de huellas dactilares y su hardware es compatible con el sistema operativo usado.
- **Teclado numérico:** se escogió un teclado Toshiba que además cuenta con dos puertos USB externos que permiten extender los de la Raspberry Pi para el momento de pruebas. El teclado permite que el usuario interactúe con la aplicación.
- **Corneta:** se determinó el uso de una corneta genérica, alimentada con 5V y entrada de audio estéreo con un cable jack de 3,5mm. La corneta

reproduce los sonidos para informar al usuario de las acciones que se realizan en la puerta.

- **Pestillo eléctrico:** por facilidad de uso y bajo costo, se adquirió un pestillo electromagnético, que al hacer pasar 12V hace que el pestillo se cierre. Al dejar de pasar la corriente, este vuelve a su estado natural. De esta forma permite a la puerta cerrarse y abrirse.
- **Circuito de apertura del pestillo eléctrico:** debido a que la fuente de energía que necesita el pestillo era mayor de la que ofrece la computadora, se debió diseñar un circuito que permita que cuando una señal de 5V (GPIO de la Raspberry Pi) es activada, habilita el paso de corriente de 12V al motor. De esta forma se abre el pestillo con el uso de la computadora.
- **Sensor de efecto Hall:** circuito que consta de un sensor de efecto Hall que detecta el magnetismo presente en la zona cercana. Este mecanismo permitirá saber el estado de la puerta. Si la puerta se encuentra cerrada, el sensor detecta el imán y no emite corriente, en cambio si la puerta está abierta, el sensor emite corriente. Esta corriente es capturada por la computadora y de esta manera conoce el estado de la puerta.
- **Iluminación para la cámara:** circuito que cuenta con un LED capaz de iluminar la zona de visión para brindarle mayor iluminación a la cámara en situaciones oscuras.

- **Transformador de 120V a 12V:** dispositivo encargado de energizar el pestillo eléctrico a través del circuito de apertura.
- **Transformadores de 120V a 5V:** dos transformadores de corriente para otorgarle energía al computador y a la corneta respectivamente.
- **Puerta de madera:** componente que contiene todos los dispositivos mencionados anteriormente.
- **Marco de la puerta:** componente encargado de sostener en pie la puerta de madera, simula marco de puerta fijo.

I.2 Diseño de circuitos electrónicos

I.2.1 Circuito de apertura del pestillo eléctrico

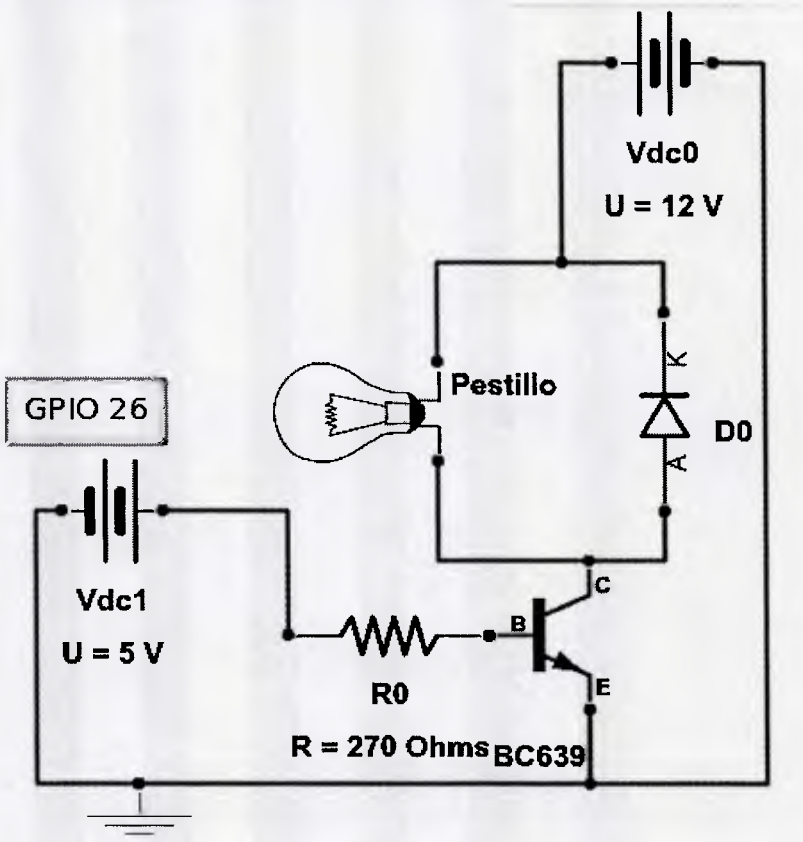


Figura 22: Circuito de apertura del pestillo eléctrico
Fuente: Elaboración propia

I.2.2 Circuito para sensor de efecto Hall

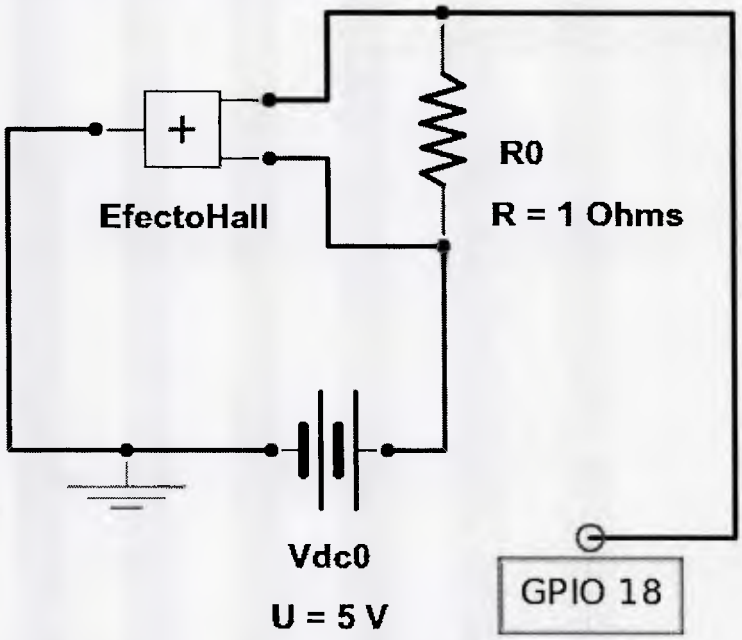


Figura 23: Circuito para sensor de efecto Hall
Fuente: Elaboración propia

I.2.3 Circuito de LED para iluminación en la puerta

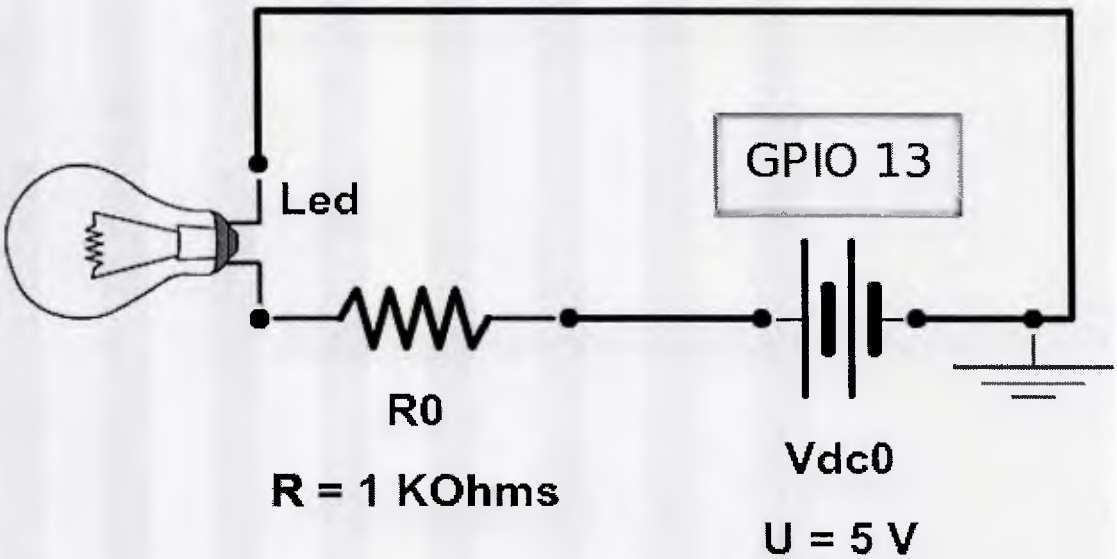


Figura 24: Circuito de LED para iluminación en la puerta
Fuente: Elaboración propia

I.3 Diagrama de cableado y distribución de componentes

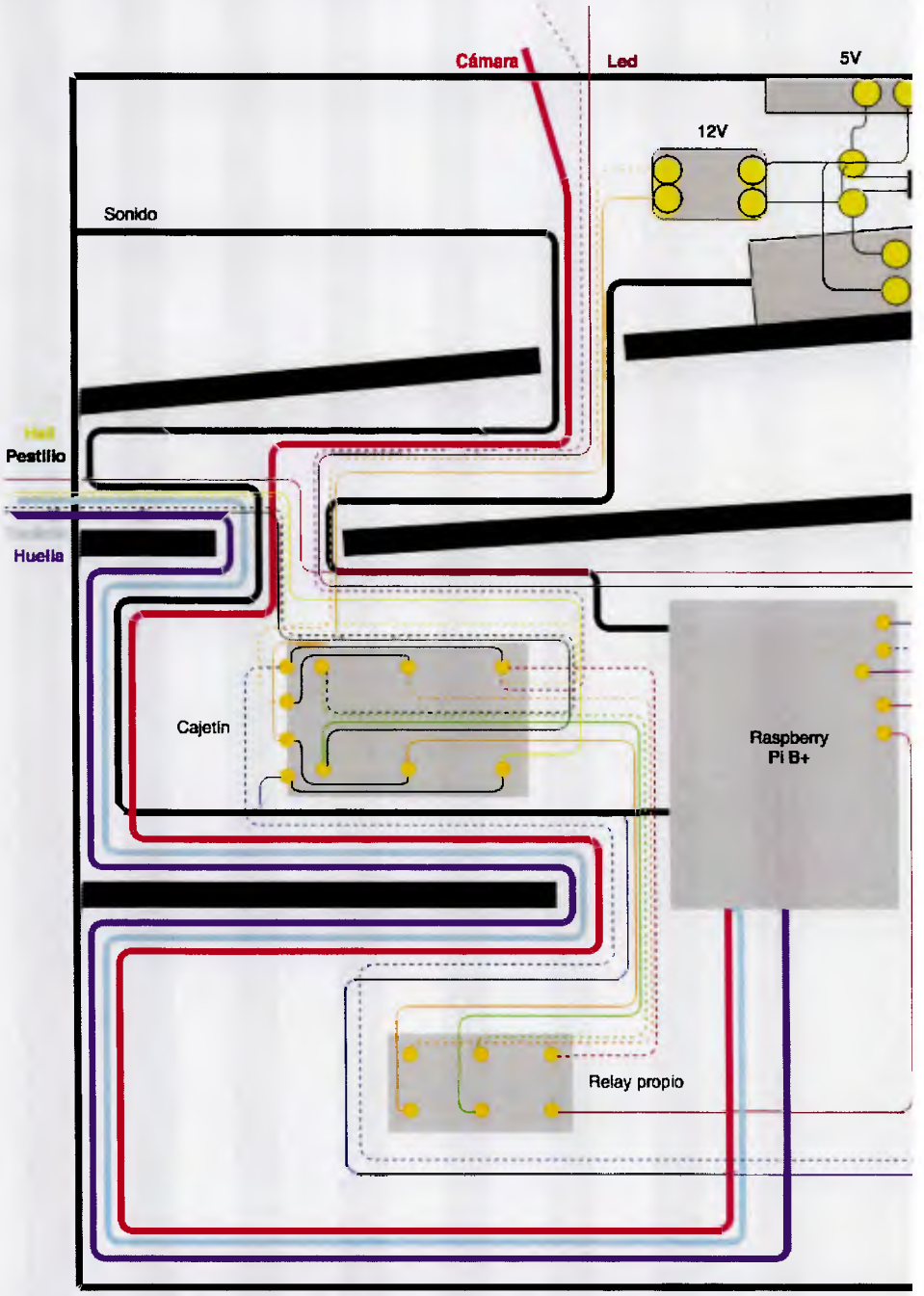


Figura 25: Diagrama de cableado y distribución de componentes
Fuente: Elaboración propia

Anexo J - Evaluación de cerraduras de seguridad

Se realizó un estudio de los diversos tipos de cerraduras existentes en el mercado, con el fin de encontrar un dispositivo que tuviese la capacidad de accionarse con el uso de una fuente eléctrica y que fuera lo más económico posible. A continuación se encuentran los tipos de cerradura que se evaluaron:

Cerradura electromagnética

La cerradura electromagnética consta de dos placas metálicas, una que se fija en la parte superior de la puerta, y otra que se instala en el marco de la puerta en el punto donde está la primera placa. La placa del marco debe estar conectada a una fuente eléctrica, la cual permite que se genere un campo magnético que mantenga las dos placas unidas. La seguridad de estas puertas tiene que ver con la fuerza magnética que se aplica entre ellas.

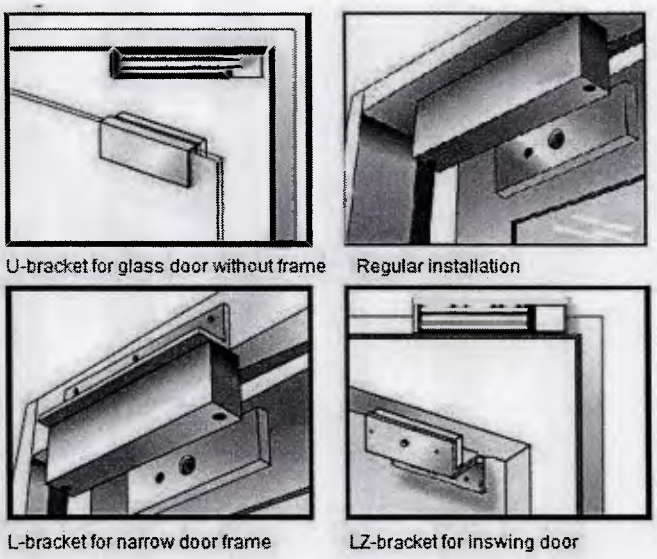


Figura 26: Cerradura electromagnética
Fuente: locknet.com

Este tipo de cerraduras se presenta en dos formas, una normalmente cerrada, que se abre al suministrarle corriente eléctrica, y otra normalmente abierta, que se cierra cuando se aplica corriente eléctrica. Los precios de estas cerraduras dependen de la fuerza aplicada. Se pueden encontrar desde 20$.

Su principal bondad es la facilidad de instalar la cerradura, ya que únicamente se deben atornillar las placas para que se mantengan en el marco y la puerta, sin necesidad de realizar aperturas. Otro beneficio es la durabilidad, al ser partes únicas que no tienen movimientos mecánicos por dentro, no sufren de desgaste.

El problema principal de esta cerradura es el alto consumo energético que presenta, debido a la corriente eléctrica constante que requiere para su funcionamiento. Además, se debe tener una fuente de poder alterna en caso de que se produzca un corte eléctrico, ya que si no es así, la cerradura pierde el magnetismo y se abre. En cuanto a los requerimientos de la puerta se tiene que la fuente de corriente se conecta desde el lado del marco y no desde la puerta.

Abrepuertas eléctrico

Este tipo de cerradura es un componente que se instala en el marco de las puertas, donde se ubica el pestillo. La función del mismo es que al pasar una corriente eléctrica, se acciona un solenoide y libera el pestillo para que pueda abrirse la puerta. El costo de este sistema de cerradura va desde los 25$.

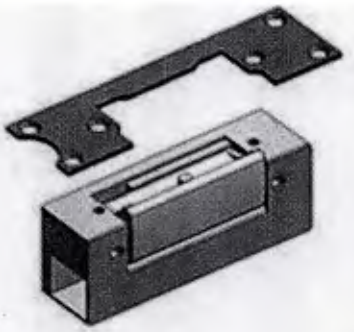


Figura 27: Abrepuertas eléctrico
Fuente: andreu.es

El factor positivo de este sistema es que si se produce un corte de luz, se puede aperturar la puerta de forma mecánica, ya sea con una llave o con la manilla de la puerta. El factor negativo es que el cableado del sistema debe realizarse a través del marco de la puerta, posterior a la instalación de la misma.

Pestillo eléctrico

Esta cerradura consta de un cilindro y una bobina, que al accionarse produce una fuerza electromagnética que hace que el cilindro se contraiga y aperture la puerta. Este tipo de cerraduras eléctricas es uno de los más económicos en el mercado, su costo va desde los 5$. Otro beneficio de este tipo de cerradura es que consume corriente únicamente cuando es accionada. En cuanto a su instalación, puede sustituir el pestillo que ya esté instalado en la puerta.

Figura 28: Pestillo eléctrico
Fuente: Amazon.com Inc.

Un factor negativo es que no se tiene forma de aperturarlo en caso que no exista energía eléctrica, lo cual puede solucionarse con una batería de 12V.

Cerradura electromecánica

Esta cerradura tiene la capacidad de aperturarse tanto mecánicamente con el uso de una llave, como con una fuente eléctrica que accione el pestillo. Esto hace que pueda funcionar de manera correcta en ocasiones que se produzcan cortes de luz. El costo de estas cerraduras es de los más altos en el mercado. Se pueden conseguir desde un mínimo de 200$.

Figura 29: Cerradura electromecánica
Fuente: Stanley Security Solutions, Inc.

El principal beneficio de esta cerradura es la posibilidad de ser usada cuando falla la energía eléctrica. Además, la mayoría de las empresas de cerraduras están elaborando este tipo de cerraduras, por lo que se convertirán en un estándar. Este es el tipo de cerradura más y en algunos casos debe ser instalada por fuera de la puerta debido a su gran tamaño.

Resultado de la evaluación

Para el prototipo de puerta diseñado, es necesario que el componente encargado de la apertura esté dentro de la puerta y no fuera de ella o en el marco, por lo que se descarto el abre puertas eléctrico.

La cerradura electromecánica cumple con todos los requerimientos para el prototipo pero el costo es muy alto en comparación con el resto, por lo que se decidió utilizar el pestillo eléctrico para simular la función de la cerradura de seguridad en el prototipo.

En caso de que el costo no sea una limitante, el tipo de cerradura que presenta las mejores características de seguridad y versatilidad es la cerradura electromecánica.