

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**DESARROLLO DE UN PROTOTIPO DE IDENTIFICACIÓN DE PERSONAL
BASADO EN TECNOLOGÍA NFC SOBRE DISPOSITIVOS MÓVILES
ANDROID PARA LA EMPRESA MIVSELLTECH S.A**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE: INGENIERO EN
SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

ESTEBAN JAVIER ORTEGA TAPIA

esteban_276@hotmail.com

DIRECTOR: Msc. Ing. RAÚL CÓRDOVA

raul.cordova@epn.edu.ec

Quito, octubre 2015

DECLARACIÓN

Yo, Esteban Javier Ortega Tapia, declaro bajo juramento que el trabajo aquí descrito es de mi autoría y no ha sido previamente presentado para ningún grado o calificación profesional; y, que se ha consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedo los derechos de propiedad intelectual correspondiente a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la ley de Propiedad Intelectual, por su reglamento y por la normatividad institucional vigente.

Esteban Javier Ortega Tapia

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por: Esteban Javier Ortega Tapia, bajo mi supervisión.

Ing. Raúl Córdova
DIRECTOR DE PROYECTO

AGRADECIMIENTO

A mi madre Albita, que con su cariño y ternura ha sabido llenar los espacios vacíos que dejan las derrotas o los malos momentos.

A mi padre Geovanny, que con sus enseñanzas ha sabido cultivar en mí principios y valores que se han fortalecido con el pasar de los años.

A cada uno de mis hermanos, Geovanny, Pancho y Ricardo, por demostrarme que una sonrisa en el rostro puede hacer olvidar cualquier problema y que unas palabras pueden ser el aliento necesario para seguir adelante.

A toda mi familia en general, por aportar con su granito de arena en cada una de las etapas de mi vida y recordarme que

A mis amigos, por permitirme formar parte de momentos memorables, por alentarme en los momentos donde creí estar derrotado y ayudarme a ver el camino hacia la victoria. Cada día transcurrido junto con las amistades, han sido días llenos de anécdotas y experiencias.

A mi director de tesis, Ing. Raúl Córdova, por compartir sus enseñanzas, consejos y experiencia en el campo profesional, por su apoyo y tiempo invertido en este proyecto.

A la Escuela Politécnica Nacional y quienes la conforman, por ayudar a materializar un sueño, por ayudar a formar ideales y especial agradecimiento por crear una identidad en cada uno de los estudiantes.

Finalmente, agradezco a la vida, por demostrarme que cada suceso viene acompañado de un propósito, por demostrarme que cada error es una oportunidad para aprender y mejorar, por enseñarme que una caída no es nada más que la causa del éxito y que el éxito surge al levantarse de una caída. No ha habido nada más grato que las sorpresas de la vida, que aunque muchas veces no las acepte, o las comprenda, o las acepte, me han llevado a ser quien soy a tener lo que tengo.

Javi

DEDICATORIA

Dedico este trabajo a mis padres: Albita y Geovanny, por haberme dado la libertad de elegir el camino, por haberme dado la libertad de recorrer dicho camino y la libertad de dejarme vivir mi camino.

A mis hermanos: Geovanny, Pancho y Ricardo, por brindarme sus consejos mientras en el recorrido de mi camino y formar parte de este.

A mis amigos, por estar presentes en los momentos indicados para con su voz guiarme a través de esta etapa universitaria y esperando, con gran dicha, que esas voces se sigan manifestando en mis etapas futuras.

A mi familia en general, quienes han sabido colaborar con un ideal, un sueño, un aliento o un consejo en cada momento de mi vida, vida que sería vacía de no ser por el apoyo incondicional que he encontrado en ellos.

Javi

Contenido

CAPÍTULO 1. Planteamiento del Problema	9
1.1. Principios de la tecnología NFC.....	9
1.1.1. INTRODUCCIÓN.....	9
1.1.2. forma de interacción de la tecnología nfc.....	10
1.1.3. Especificaciones de NFC.....	10
1.1.4. usos comunes	11
1.1.5. Principio de funcionamiento.....	12
1.1.6. Emulación de la tarjeta basada en host	14
1.1.7. Android y NFC	15
1.1.8. Limitaciones NFC	15
1.2. Análisis y Selección de la Metodología de Desarrollo	15
1.2.1. introducción	15
1.2.2. Metodologías Ágiles	16
1.2.3. Análisis comparativo.....	19
1.2.4. Necesidades del proyecto.....	20
1.2.5. rad (rapid application development).....	20
1.2.6. Selección de la metodología.....	23
1.3. Protocolos y Herramientas de Software y Hardware.....	24
1.3.1. Protocolos	24
1.3.2. Software	26
1.3.3. Hardware.....	26
1.4. Análisis de Dispositivos involucrados.....	27
1.4.1. Moto X (TAG NFC)	27
1.4.2. ACR 122 (Lector NFC)	27
CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL PROTOTIPO	29
2.1. requerimientos.....	29
2.1.1. Registro de empresa	29
2.1.2. Actualización de empresa.....	29
2.1.3. Registro de empleados.....	29
2.1.4. Actualización de empleados	29
2.1.5. Monitorear accesos	30
2.1.6. Registro de accesos	30
2.1.7. Actualización de accesos	30
2.1.8. Registro de áreas	30
2.1.9. Actualización de área	30

2.1.10.	Registro de lectores.....	31
2.1.11.	Actualización de lectores	31
2.2.	Historias de usuario	31
2.2.1.	Registro de empresas.....	31
2.2.2.	Actualización de empresas	32
2.2.3.	Cambio de estado	32
2.2.4.	Registro de empleados.....	33
2.2.5.	Cambio de estado de empleado	33
2.2.6.	Actualización de empleado	34
2.2.7.	Registro de lector	34
2.2.8.	Actualización de lector	35
2.2.9.	eliminar lector	35
2.2.10.	Registrar Acceso	35
2.2.11.	Actualización de accesos	36
2.2.12.	Eliminación de accesos	36
2.2.13.	Enlace con lectores	37
2.2.14.	Captura de datos	37
2.3.	Plan de Entregas	37
2.4.	Priorización.....	38
2.5.	Plan de entregas	39
2.6.	Tarjetas CRC.....	39
2.6.1.	Empleado	40
2.6.2.	empresa	40
2.6.3.	área	40
2.6.4.	lector	40
2.6.5.	Monitor	41
2.6.6.	TAG.....	41
2.7.	análisis	41
2.7.1.	situación actual.....	42
2.8.	diseño.....	42
2.8.1.	Entorno de desarrollo	42
2.8.2.	arquitectura	44
2.8.3.	Diseño de interfaces.....	46
2.8.4.	Instalación y configuración de software base.....	58
CAPÍTULO 3. implemeNtación, pruebas y análisis de resultados		69
3.1.	implementación	69

3.1.1.	Instalación del aplicativo de control de accesos.....	70
3.1.2.	Base de datos POSTGRESQL 9.3	70
3.1.3.	archivo de configuracion	71
3.1.4.	Autorización.....	73
3.2.	pruebas	73
3.2.1.	pruebas funcionales.....	73
3.3.	análisis de resultados	81
CAPÍTULO 4. conclusiones y recomendaciones.....		90
4.1.	conclusiones.....	90
4.2.	recomendaciones	90
BIBLIOGRAFÍA.....		91
GLOSARIO		94

CAPÍTULO 1. PLANTEAMIENTO DEL PROBLEMA

1.1. PRINCIPIOS DE LA TECNOLOGÍA NFC

1.1.1. INTRODUCCIÓN

En la actualidad, el avance de la tecnología ha hecho que tomemos ciertos atajos para resolver problemas u obtener resultados. La simplicidad que los dispositivos tecnológicos presentan, tanto en usabilidad como en adquisición, ha tenido un masivo impacto en la sociedad, principalmente en lo cultural.

La interacción de los dispositivos electrónicos y los usuarios se ha venido llevando a cabo de manera cada vez más natural. las tareas cotidianas se efectúan de una manera completamente transparente con respecto a los dispositivos electrónicos. Este paradigma es conocido como Computación Ubicua que se define como el nivel de interacción entre los ordenadores y los usuarios.

La aparición de dispositivos inteligentes como los smartphones, tablets, reproductores digitales, etc. ha permitido que la manera de laborar, procesar y compartir información cambie. Hay que comprender que la aparición y el éxito de estos dispositivos móviles se debe en gran parte a la facilidad en la prestación de servicios de comunicación como las redes y los sistemas distribuidos. (Vedat, Karen, & Ozdenizci, 2013)

La tecnología NFC simplifica la comunicación entre dispositivos en un entorno reducido. Esta tecnología ya ha sido empleada por muchos años en tarjetas de crédito, tarjetas electromagnéticas, tickets o tarjetas de acceso a instalaciones y para control de personal, pero ha pasado desapercibida para los usuarios y a veces, hasta incomprendida. Actualmente se la ha incorporado en los dispositivos móviles, lo que está provocando que el uso de los dispositivos ya señalados tienda a desaparecer.

El entorno de la tecnología NFC fue diseñado a partir del desarrollo de otras tecnologías como la comunicación inalámbrica, los dispositivos móviles, aplicaciones móviles y tecnologías de tarjetas inteligentes. De igual manera, los servicios web y cloudcomputing (computación en la nube) también ayudaron a su propagación.

La tecnología NFC puede ser identificada como la combinación de tecnologías de identificación contactless y tecnologías de interconexión. NFC es una tecnología que opera entre dos dispositivos en un rango corto de comunicación.

1.1.2. FORMA DE INTERACCIÓN DE LA TECNOLOGÍA NFC

La manera de interactuar de los dispositivos ha evolucionado a lo largo de los años. Los usuarios de computadoras de escritorio o portátiles han tenido a lo largo de los años dispositivos de entrada, como el mouse o teclado y dispositivos de salida, como el monitor.

Hace pocos años, las pantallas táctiles cambiaron la forma en la cual los usuarios podían interactuar con los dispositivos, eliminando en cierta manera a los dispositivos de entrada ya mencionados. De igual forma, la manera de interactuar con el entorno ha evolucionado dramáticamente con las redes móviles, al aparecer los sistemas de mensajería, los sistemas de conexión inalámbricas (WiFi - Bluetooth), que permitieron acceder a la información de una manera más ágil pero insegura en muchos casos. (Roland, 2015)

Ahora, una nueva forma de interactuar con los dispositivos externos se ha introducido en los dispositivos móviles: la tecnología NFC, cuya evolución se muestra en la Figura 1.1.

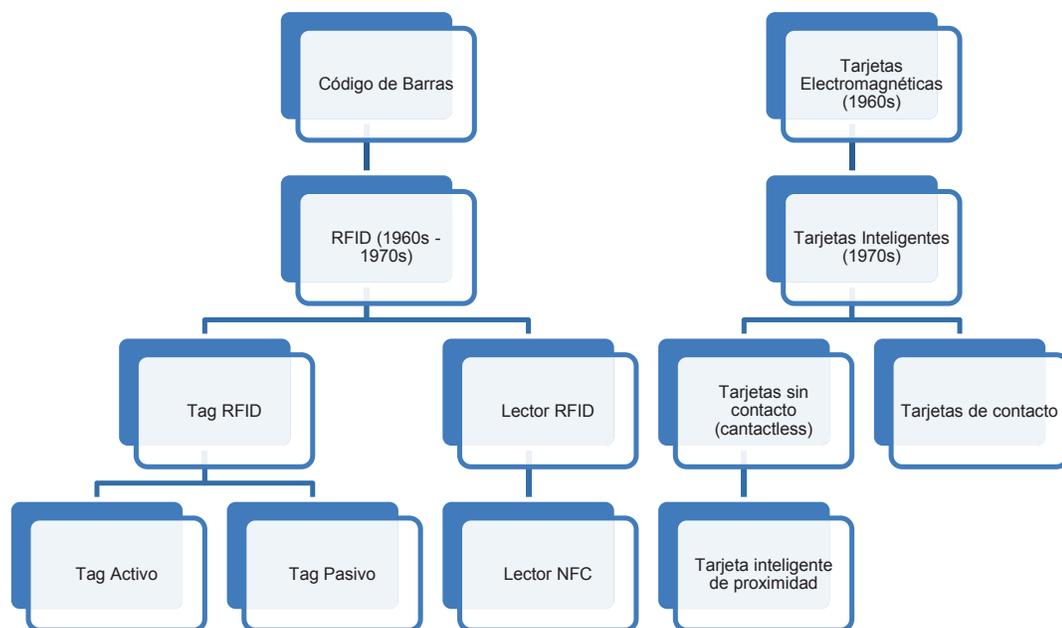


Figura 1.1 Evolución de la tecnología NFC

1.1.3. ESPECIFICACIONES DE NFC

NFC es una tecnología de comunicación inalámbrica de corto alcance, similar a la tecnología infrarroja o Bluetooth. La transmisión de datos entre dos dispositivos NFC se produce en el rango de frecuencia de 13,56 MHz. La comunicación gama se limita a una proximidad muy cercana, generalmente de hasta 10 cm . La máxima velocidad de comunicación se limita a 424 kbps.

NFC siempre forma conexiones punto a punto entre 2 dispositivos. Para este tipo de conexión, esta tecnología es más rápida que otras tecnologías inalámbricas, ya que el emparejamiento es más simple, pero ofrece tasas de transferencia más bajas en comparación con sus competidores.

La comparación con otras tecnologías inalámbricas se muestra en la Figura 1.2. La configuración de conexión entre dos dispositivos compatibles con NFC es muy intuitiva, permitiendo una conexión automática cuando se colocan los dispositivos cercanos entre sí. (Finkenzeller, 2010)

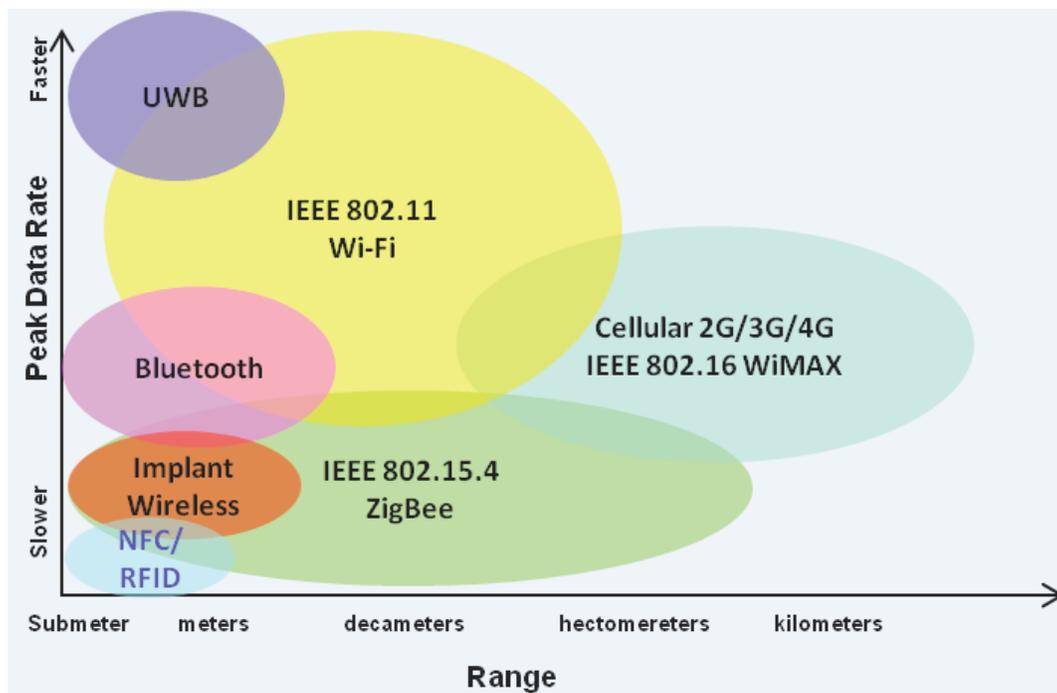


Figura 1.2 Rango de tecnologías inalámbricas(CNXSoft)

NFC se define como una tecnología de conectividad basada en estándares. Ha surgido como una nueva tecnología, construida en gran medida por tecnologías y estándares ya existentes. Se incluyen Radio identificación de frecuencia (RFID), tarjetas inteligentes y tarjetas sin contacto (Contactless).

1.1.4. USOS COMUNES

La forma más común en que la tecnología NFC está siendo usada es para ajustar eficientemente una serie de configuraciones de sistemas basados en la llegada o salida de los lugares comúnmente visitados.

Organismos como SITE (Smart Integrated Ticketing for Europe), busca incorporar herramientas tecnológicas sobre zonas turísticas y sobre zonas de interés para la población, la herramienta tecnológica que más auge ha tenido para esta organización es NFC.

A continuación se detallan ciertos usos que se han encontrado para NFC y se han puesto en práctica en muchos países europeos.

- En el hogar

Sobre muchos hogares o sitios turísticos se han colocado etiquetas NFC en los accesos principales para permitir acceder a Wi-Fi de manera mucho más rápida, controlar el sistema de iluminación. El uso de NFC sobre los hogares ha permitido que se puedan configurar normas de seguridad como activación de alarmas, activación de cerraduras, etc. Estas configuraciones puedan ser activadas o desactivadas, dependiendo del caso, empleando únicamente teléfonos inteligentes.

- En los vehículos

Sobre muchos vehículos se ha empleado esta tecnología para almacenar configuraciones especiales como colocación del asiento del piloto, posición de los retrovisores, volumen predeterminado, etc. Estas tareas han podido ser parametrizadas de manera casi personalizadas con ayuda de esta tecnología.

- En el trabajo

En el campo laboral la tecnología NFC ha avanzado a pasos agigantados ya que los empresarios así como las organizaciones han apoyado en gran parte a la incorporación de esta tecnología en las empresas para obtener beneficios que ayudan a la mejora de la productividad de los empleados. La implementación de sistemas de control de acceso así como de pagos de cafetería a través de NFC, son comúnmente usados. Entrega de notificaciones de manera masiva ha sido solucionado gracias a la incorporación de NFC.

1.1.5. PRINCIPIO DE FUNCIONAMIENTO

Teniendo en cuenta una comunicación entre dos dispositivos NFC, es necesario distinguir si cada dispositivo posee su propia fuente de energía y si es capaz de generar su propio campo de Radio Frecuencia (RF) o tomar energía de un campo generado por otro dispositivo. Un dispositivo que genera su propio campo de RF es conocido como dispositivo activo, y aquellos dispositivos que no tienen la facultad de generar un campo de radio frecuencia son conocidos como dispositivos pasivos.

Las etiquetas NFC siempre han sido dispositivos pasivos porque no tienen su propia fuente de energía, sin embargo, los dispositivos con NFC, como un teléfono móvil o un lector de NFC, pueden actuar así como un dispositivo activo o pasivo. Además de las interfaces individuales pueden desempeñar dos funciones diferentes, el iniciador (dispositivo maestro) o de destino (dispositivo esclavo). La comunicación NFC siempre se inicia por el iniciador.

Las interfaces de comunicación que operan en el modo de petición y respuesta. El iniciador envía un mensaje de solicitud y el objetivo sólo responde enviando un mensaje de respuesta. Sólo el dispositivo activo puede actuar como un iniciador y el dispositivo pasivo sólo puede actuar como un objetivo en la comunicación que entrega una respuesta específica y programada.

1.1.5.1. Modo pasivo¹

En este modo, el iniciador induce un campo electromagnético de alta frecuencia (RF) para transmitir datos al objetivo (lector). La amplitud de la señal portadora se modula de acuerdo a los datos. Después que los datos se transmiten, la señal continúa a ser emitida de una manera no modulada. (Vedat, Karen, & Ozdenizci, 2013)

1.1.5.2. Modo activo²

En este modo, el dispositivo Iniciador activa su transmisor y genera un campo de RF que induce la tensión en el bucle de antena del segundo dispositivo. Cuando esto se detecta en el segundo dispositivo, cambia al modo de objetivo y procesa la señal. La tarjeta pasiva responde al generar la misma RF como el iniciador, sólo la dirección de transmisión se invierte.

Los dispositivos NFC operan bajo tres modos diferentes: el lector / grabador, peer-to-peer o por modo de emulación de tarjeta (modo del cual nos enfocaremos). El caso de uso y arquitectura técnica subyacente es diferente para cada uno de estos modos:

- El modo de lector / escritor:

En el modo de lectura - escritura, el dispositivo NFC funciona como un lector estándar de RFID y es capaz de leer etiquetas NFC pasivas, o en general cualquier tarjeta o dispositivo NFC compatible con las normas ISO / IEC 14443 y FeliCa.

- El modo Peer – to – peer (P2P):

En el modo P2P, el dispositivo NFC tiene una conexión bidireccional con otro dispositivo NFC en el mismo modo con el fin de intercambiar datos. Esto puede ser utilizado para el intercambio de pequeñas cantidades de datos entre teléfonos (por ejemplo, direcciones URL), el intercambio de datos de contacto o bootstrappingWiFi. Este modo está estandarizado en la norma ISO 18092 / IEC.

- El modo de emulación de la tarjeta:

¹<http://computerhoy.com/noticias/life/nfc-que-es-que-sirve-esta-conexion-inalambrica-24207>

²<https://mundonfc.wordpress.com/2012/02/09/como-funciona-el-nfc/>

En este modo, el dispositivo con NFC emula tarjetas inteligentes (ISO / IEC 14443 o la tarjeta FeliCa) y actúa como un dispositivo pasivo. La emulación es transparente y para lectores externos, tales dispositivos aparecen como una tarjeta inteligente tradicional. Esto asegura la compatibilidad con las tarjetas inteligentes existentes y por lo tanto la emulación se utiliza principalmente para los pagos sin contacto y venta de entradas. El lector externo normalmente se comunica con una aplicación de un elemento de seguridad, ya que proporciona una mayor seguridad para el funcionamiento de las aplicaciones críticas de seguridad. (Vedat, Karen, & Ozdenizci, 2013)

1.1.6. EMULACIÓN DE LA TARJETA BASADA EN HOST

Hablando sobre el modo de emulación de la tarjeta, hay que distinguir entre dos variantes básicas:

- Emulación de la tarjeta con un elemento de seguridad

En este caso, el programa de emulación de tarjetas se está ejecutando en el elemento seguro, lo que se comunica directamente con el controlador NFC, sin pasar por el Android sede de la CPU. Esto se utiliza normalmente para las aplicaciones de pago.

- Emulación de tarjetas basado en host (HCE)

También conocida como la emulación de tarjeta de Software, en este modo los datos recibidos en el modo de emulación de la tarjeta son transferidos a la CPU del host, donde las aplicaciones de Android se ejecutan directamente. Por lo que no existe la presencia de ningún elemento de seguridad involucrado en la comunicación.

Previo a nivel de API 19 (AndroidKitKat 4.4) , no había manera de cambiar la interfaz NFC al modo basado en host en Android. Desde la versión de Android 4.4 KitKat , esto es ahora parte de la API estándar. Android sólo es compatible con la emulación de tarjetas basados en la ISO -DEP en este momento, obligatoriamente basado en la tecnología NFC -A y opcionalmente también NFC -B.

La arquitectura de HCE en Android se basa en componentes de servicio (denominados Servicios HCE), que funcionan como aplicaciones instaladas emulando una tarjeta. Cada servicio se identifica por un ID de aplicación (AID), definida por la norma ISO / IEC 7816/4. Cuando existe un contacto entre el dispositivo y un lector de NFC, se selecciona una aplicación adecuada sobre la base de la ayuda solicitada. El HCE incluso puede coexistir con otros métodos de aplicación de emulación de la tarjeta. La selección de la ruta de enrutamiento se basa en la ayuda que figura en el comando SELECT APDU. (Finkenzeller, 2010)

1.1.7. ANDROID Y NFC

Android es un nombre general que se utiliza para el sistema operativo para móviles, dispositivos con este sistema y toda la plataforma móvil. Android es un sistema operativo de código abierto desarrollado específicamente para dispositivos móviles (smartphones, tablets, televisores inteligentes, etc) equipados con pantallas táctiles y que tienen limitados recursos computacionales. Android es propiedad de OpenHandsetAlliance, liderada por Google, y es de origen completamente abierto. En esencia, se trata de una versión modificada de Linux con interfaz integrada de programación con Java. El entorno de desarrollo de aplicaciones incluye un compilador, depurador, emulador y la máquina virtual de Java conocida como la máquina virtual Dalvik (DVM). DVM es diferente de la de Oracle JVM oficial y que está desarrollada de forma independiente. (Vedat, Karen, & Ozdenizci, 2013)

Aplicaciones para Android están escritas en Java y se compilan en archivos ejecutables Dalvik. Tales archivos de aplicación se ensamblan en formato apk, que se utiliza como un contenedor de aplicación universal. Android también proporciona su propia biblioteca de clases de Java, no alineado con el estándar Java SE o Java ME. La biblioteca proporciona un API para las funciones del sistema subyacentes y funcionalidades comunes, que se superponen parcialmente con la library6 Java SE.

1.1.8. LIMITACIONES NFC

La tecnología NFC es sorprendentemente limitada para los desarrolladores de aplicaciones en la plataforma Android, en comparación con otras tecnologías inalámbricas existentes. En primer lugar, no existe una manera de utilizar la funcionalidad peer – to – peer, aparte de la tecnología AndroidBean. AndroidBean se basa en el protocolo SNEP, que permite el intercambio transparente de datos NDEF. Desafortunadamente, no hay una API para el uso directo del protocolo SNEP en las versiones más antiguas del sistema operativo Android. (Inside Secure, 2012)

1.2. ANÁLISIS Y SELECCIÓN DE LA METODOLOGÍA DE DESARROLLO

1.2.1. INTRODUCCIÓN

El desarrollo de aplicaciones móviles sufre prácticamente los mismos problemas que la gran mayoría de desarrollos de software. Aunque hay que tener en cuenta sus principales peculiaridades como la corta duración de sus desarrollos, la gran competencia del sector que obliga a una constante innovación, los cambios frecuentes en la plataforma de desarrollo y en el hardware o la simplicidad de algunas aplicaciones. Todo esto influye a la hora de elegir una metodología concreta de desarrollo.

La mayor parte de los proyectos de desarrollo de software se llevan a cabo por equipos de desarrolladores pequeños que requieren de un método de desarrollo común para organizar sus tareas, ya sean de una forma ágil o de un modelo más estático y predictivo.

Desde hace años, con el creciente desarrollo tecnológico cada vez más personas acceden a dispositivos móviles, los cuales han evolucionado constantemente en capacidades de servicio, ofreciendo funciones de comunicación y procesamiento de datos, que van mucho más allá que las simples llamadas telefónicas o ejecución de aplicaciones básicas. Las particularidades específicas de un entorno móvil incluyen un alto nivel de competitividad, un tiempo necesariamente corto de entrega de aplicaciones, más la dificultad adicional que implica la identificación de las partes interesadas (stakeholders) y sus requerimientos. Los equipos de desarrollo deben enfrentar los desafíos de este entorno cambiante, con frecuentes modificaciones en las necesidades y expectativas de los clientes.

1.2.2. METODOLOGÍAS ÁGILES

Las metodologías ágiles han demostrado ser muy beneficiosas al ayudar a los equipos de software de TI a la hora de entregar a tiempo software de alta calidad que se ajuste a las necesidades de los usuarios finales. Los equipos de software quieren trabajar con metodologías ágiles porque necesitan un proceso que pueda responder de manera eficiente a los cambios en los productos en desarrollo. Las metodologías ágiles permiten una mayor flexibilidad que las metodologías tradicionales de desarrollo, que se bloquean muy pronto en los detalles del proyecto y son menos capaces de ajustarse a las cambiantes necesidades de los usuarios, del mercado y de los desafíos imprevistos que plantea la tecnología.

Las metodologías ágiles son una serie de técnicas para la gestión de proyectos que han surgido como contraposición a los métodos clásicos o tradicionales de gestión.

Todas las metodologías que se consideran ágiles cumplen con el manifiesto ágil que no es más que una serie de principios que se agrupan en 4 valores:

1. Los individuos y su interacción, por encima de los procesos y las herramientas.
2. El software que funciona, frente a la documentación exhaustiva.
3. La colaboración con el cliente, por encima de la negociación contractual.
4. La respuesta al cambio, por encima del seguimiento de un plan.

Inicialmente se asocia a las metodologías ágiles con falta de documentación o control sobre el proyecto, sin embargo, lo que se busca es totalmente distinto. Lo que se desea es minimizar el impacto de las tareas que no son relevantes para conseguir el objetivo del proyecto. Se pretende aumentar la eficiencia de

las personas involucradas en el proyecto y, como resultado de ello, minimizar los costos del proyecto. (CASEMaker Inc., 2000)

1.2.2.1. Principales Metodologías Ágiles

Uno de los principales focos de aplicación de las metodologías ágiles son los proyectos tecnológicos. Cada una de ellas tienen sus fortalezas y sus debilidades. En cada proyecto podemos adoptar una, o varias, en función de las características del propio proyecto y del equipo.

Aunque los creadores e impulsores de las metodologías ágiles más populares han suscrito el manifiesto ágil y coinciden con los principios, cada metodología tiene características propias y hace hincapié en algunos aspectos más específicos. A continuación se resumen dichas metodologías ágiles.

1.2.2.1.1. *Extreme Programming (XP)*³

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen ambiente de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. Kent Beck, el padre de XP, describe la filosofía de XP en sin cubrir los detalles técnicos y de implantación de las prácticas. Posteriormente, otras publicaciones de experiencias se han encargado de dicha tarea. A continuación presentaremos las características esenciales de XP organizadas en los tres apartados siguientes: historias de usuario, roles, proceso y prácticas.

1.2.2.1.2. *SCRUM*⁴

Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, denominadas *sprints*, con una duración de 30 días. El resultado de cada *sprint* es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto. Éstas son las

³<https://procesosdesoftware.wikispaces.com/METODOLOGIA+XP>

⁴<http://proyectosagiles.org/que-es-scrum/>

verdaderas protagonistas, especialmente la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración.

1.2.2.1.3. *Crystal Methodologies*⁵

Se trata de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo (de ellas depende el éxito del proyecto) y la reducción al máximo del número de artefactos producidos. Han sido desarrolladas por AlistairCockburn. El desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en equipo definidas. Estas políticas dependerán del tamaño del equipo, estableciéndose una clasificación por colores, por ejemplo Crystal Clear (3 a 8 miembros) y Crystal Orange (25 a 50 miembros).

1.2.2.1.4. *Dynamic Systems Development Method (DSDM)*⁶

Define el marco para desarrollar un proceso de producción de software. Nace en 1994 con el objetivo el objetivo de crear una metodología RAD unificada. Sus principales características son: es un proceso iterativo e incremental y el equipo de desarrollo y el usuario trabajan juntos. Propone cinco fases: estudio viabilidad, estudio del negocio, modelado funcional, diseño y construcción, y finalmente implementación. Las tres últimas son iterativas, además de existir realimentación a todas las fases.

1.2.2.1.5. *Adaptive Software Development (ASD)*⁷

Su impulsor es JimHighsmith. Sus principales características son: iterativo, orientado a los componentes software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda desarrollan las características y finalmente en la tercera se revisa su calidad, y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo.

1.2.2.1.6. *Feature-DrivenDevelopment (FDD)*⁸

Define un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software. Sus impulsores son Jeff De Luca y Peter Coad.

⁵http://ingenieriadesoftware.mex.tl/59189_Metodologia-Crystal.html

⁶http://ingenieriadesoftware.mex.tl/52827_DSDM.html

⁷http://ingenieriadesoftware.mex.tl/61154_ASD.html

⁸http://ingenieriadesoftware.mex.tl/61162_FDD.html

1.2.2.1.7. *Lean Development (LD)*⁹

Definida por Bob Charette's a partir de su experiencia en proyectos con la industria japonesa del automóvil en los años 80 y utilizada en numerosos proyectos de telecomunicaciones en Europa. En LD, los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. Su principal característica es introducir un mecanismo para implementar dichos cambios.

1.2.3. ANÁLISIS COMPARATIVO

Una vez que se han listado las metodologías ágiles más relevantes, es necesario compara dichas metodologías con el fin de seleccionar la más adecuada para ser aplicada en este proyecto. En la Tabla 1.1 se compara las distintas aproximaciones ágiles en base a los siguientes parámetros: vista del sistema como algo cambiante, tener en cuenta la colaboración entre los miembros del equipo y características más específicas de la propia metodología como son simplicidad, excelencia técnica, resultados, adaptabilidad, etc. También incorpora como referencia no ágil el Capability Maturity Model (CMM).

Tabla 1.1 Ranking de agilidad

	CMM	ASD	Crystal	DSDM	FDD	LD	SCRUM	XP
Sistema como algo cambiante	1	5	4	3	3	4	5	5
Colaboración	2	5	5	4	4	4	5	5
Características metodología (CM)								
Resultados	2	5	5	4	4	4	5	5
Simplicidad	1	4	4	3	5	3	5	5
Adaptabilidad	2	5	5	3	3	4	4	3
Excelencia técnica	4	3	3	4	4	4	3	4
Prácticas de colaboración	2	5	5	4	3	3	4	5

⁹http://www.ecured.cu/Lean_Development

Media CM	2.2	4.4	4.4	3.6	3.8	3.6	4.2	4.4
Media total	1.7	4.8	4.5	3.6	3.6	3.9	4.7	4.8

Como se observa en la Tabla 1.1, todas las metodologías ágiles tienen una significativa diferencia del índice de agilidad respecto a CMM y entre ellas destacan ASD, Scrum y XP como las más ágiles.

1.2.4. NECESIDADES DEL PROYECTO

El proyecto debe de cumplir con ciertas características que se mencionan a continuación:

Tiempo: el proyecto debe de ser presentado en un tiempo corto.

Equipo de trabajo: el proyecto está formado por dos personas, razón por la cual es necesario simplificar las tareas y emplear herramientas CASE que ayuden en la elaboración del proyecto.

Interacción con el cliente: el cliente forma parte crucial del proyecto, razón por la cual, este puede ser considerado como un miembro activo en el proyecto.

Plataforma: el proyecto integra plataformas de escritorio así como móviles, por esta razón es necesario emplear una metodología que se acople a un entorno cambiante y evolutivo.

Flexible a cambios: el proyecto puede verse afectado por muchos cambios en lo que se refiere a plataforma, razón por la cual es necesario buscar una metodología capaz de manejar dichos cambios que puedan presentarse.

Entrega: es necesario entregar un prototipo funcional en el menor tiempo posible.

1.2.5. RAD (RAPID APPLICATION DEVELOPMENT)

Según James Martin, "el desarrollo Rápido de Aplicaciones (RAD) es una metodología diseñada para garantizar un desarrollo más rápido para obtener resultados de mejor calidad a los comúnmente obtenidos con las metodologías de desarrollo tradicionales"¹⁰.

RAD es una metodología que permite desarrollar sistemas de una manera rápida y de calidad reduciendo de gran manera los costos de desarrollo y de mantenimiento que en la mayoría de casos son factores importantes para el fracaso de proyectos de desarrollo de software.

¹⁰Rapid ApplicationDevelopment - James Martin

RAD es una metodología para la comprensión de los análisis, el diseño, la construcción y fases de prueba. Esto tiene una serie de ventajas sobre la modelo de desarrollo secuencial tradicional. (Martin, 1990)

Proyectos RAD típicamente cuentan con un mínimo de personal formados principalmente por desarrolladores, usuarios finales y recursos técnicos. Los equipos pequeños, combinado con ciclos de desarrollo cortos, optimización de la velocidad, unidad de visión y propósito, permiten tener una gestión más sencilla de los proyectos.

En el desarrollo rápido de aplicaciones se da énfasis a la obtención de un prototipo funcional de una aplicación para posteriormente ir mejorándola incluyendo más funcionalidades y complejidad según los requerimientos y necesidades de los usuarios. (CASEMaker Inc., 2000)

1.2.5.1. Diseño del modelo RAD

RAD distribuye el diseño, el análisis, el desarrollo y las fases de pruebas en ciclos de desarrollo iterativos cortos. A continuación se detallan las fases presentes en RAD:

- **Modelamiento del negocio**

Un análisis completo del negocio se lleva a cabo para encontrar la información vital para los negocios, la forma en que se puede obtener, cómo y cuándo la información debe ser procesada y cuáles son los factores que impulsan el flujo exitoso de la información.

- **Modelamiento de Datos**

La información obtenida en la fase de modelado de negocios se revisa y se analiza para formar un conjunto de objetos de datos vitales para el negocio. Los atributos de todos esos conjuntos de datos se identifican y se definen. La relación entre estos objetos de datos son establecidas y definidas en detalle.

- **Modelamiento de procesos**

Los conjuntos de objetos de datos definidos en la fase de modelado de datos son empleados para establecer el flujo de información empresarial necesaria para lograr los objetivos específicos del negocio.

- **Desarrollo de la aplicación**

El sistema es codificado

- **Pruebas y Mejoras**

Dada la naturaleza de la industria del software, está enfrenta el enorme reto de mantenerse al ritmo de las necesidades del mercado que constantemente

evolucionan, debido a factores como la competencia y la globalización. Esto hace que la brecha entre la liberación de los productos de software y su comercialización se reduzca cada vez más. Esta reducción en el tiempo de liberación ha ocasionado que el tiempo de ejecución de las pruebas sea sacrificado, lo cual es un riesgo para los proveedores de software así como para los usuarios finales, marcando con ello una dinámica de puesta en producción muy acelerada.

RAD ha surgido con el fin de generar sistemas de alta calidad y a la vez, con una reducción y simplificación de tareas. Bajo este enfoque, las pruebas de software han tomado un papel crucial, dada la necesidad de realizar pequeñas liberaciones funcionalmente estables, basándose en lo que es conocido como el testing ágil.

El tiempo total de las pruebas se reduce en el modelo RAD ya que los prototipos son probados de forma independiente en cada iteración. Estas pruebas son ejecutadas conjuntamente con los usuarios finales, con el fin de que las pruebas puedan ser llevadas a cabo conjuntamente con procesos de capacitación e introducción a la operación de los sistemas.

Este acoplamiento de las pruebas con las demás tareas, han permitido que la puesta en ejecución de los sistemas se la realice de manera más dinámica. (Martin, 1990)

1.2.5.2. Testing ágil

En general, las pruebas llevadas a cabo son diseñadas en base a requerimientos estáticos, calificativo que muchas veces es puesto por los propios testers o desarrolladores. Este diseño hace que las pruebas sean aplicables únicamente sobre funcionalidades estáticas poco evolutivas. Un cambio en el proceso de ingeniería de requerimientos, fácilmente puede ocasionar el fracaso de las pruebas.

Es imprescindible comprender que la evolución de los requerimientos durante el proceso de ingeniería de requerimientos y después de que un sistema esté en uso, es inevitable. Durante el proceso de recopilación de requerimientos, existe una fase de retroalimentación de la información, fase en la cual se comprenden de mejor manera los procesos del negocio y donde existe una ligera o gran involucración del área de desarrollo en las actividades de la organización.

El testing ágil refiere un enfoque dinámico de las pruebas de software. Dicho enfoque supone que los requerimientos no son estables sino que se incrementan de forma continua o se encuentran en constante cambio, de acuerdo a las necesidades expresadas por el cliente. Así mismo, el cliente también juega un rol muy importante manteniendo una comunicación cercana y continua con el equipo de desarrollo. La obtención de sus necesidades

(requerimientos) no se considera como una fase separada del desarrollo del software, sino una parte integral del mismo.

Este dinamismo en las pruebas, permite tener la posibilidad de acoplar tareas dentro de las pruebas con el fin de optimizar tiempo y recursos. Las tareas comúnmente acopladas son la capacitación y obtención de feedback.

Ya sin requerimientos detallados y sin un proceso de control de cambios que mantengan a los requerimientos estáticos, la habilidad más importante es la de tomar decisiones acerca del trabajo que se ejecutará. En un proyecto con cambios continuos, es importante tomar buenas decisiones respecto a lo que se prueba, y cuándo se prueba.

Hay que recalcar que los cambios continuos que se pueden dar durante un proyecto, no necesariamente se deben a una mala recopilación de los requerimientos, factores como el entorno social, legal y organizacional afectan de gran manera a la evolución de los requerimientos y de sus pruebas. (Balaguera, 2013)

1.2.5.3. Ventajas

- I. Los cambios pueden ser fácilmente ajustados al modelo.
- II. El progreso puede ser cuantificado con mayor rapidez y exactitud.
- III. Se puede lograr aumentar la productividad con muy poco personal.
- IV. Reducción en el tiempo de desarrollo y costos de producción.
- V. Si el modelo es correctamente aplicativo, logra impactar en los usuarios finales de una manera muy positiva al ver resultados reales en corto tiempo.

1.2.5.4. Desventajas

- I. La aplicación de este modelo exige que el personal perteneciente al grupo de desarrollo tenga habilidades muy desarrolladas
- II. Aplicable únicamente a sistemas modulares.
- III. Es sumamente frágil en los tiempos propuestos.

1.2.6. SELECCIÓN DE LA METODOLOGÍA

No existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software. Toda metodología debe ser adaptada al contexto del proyecto, esto implica recursos técnicos y humanos, tiempo de desarrollo, tipo de sistema, etc. Históricamente, las metodologías tradicionales han intentado abordar la mayor cantidad de situaciones de contexto del proyecto, exigiendo un esfuerzo considerable para ser adaptadas, sobre todo en proyectos pequeños y con requisitos muy cambiantes. Las metodologías ágiles ofrecen una solución casi a medida para una gran cantidad de proyectos que tienen estas características. Una de las cualidades más destacables en una metodología ágil es su sencillez, tanto en su aprendizaje como en su

aplicación, reduciéndose así los costos de implantación en un equipo de desarrollo. Esto ha llevado hacia un interés creciente en las metodologías ágiles, sin embargo, hay que tener presente una serie de inconvenientes y restricciones para su aplicación, tales como:

- Están dirigidas a equipos pequeños o medianos (Beck sugiere que el tamaño de los equipos se limite de 3 a 20 como máximo, otros dicen no más de 10 participantes)
- El entorno físico debe ser un ambiente que permita la comunicación y colaboración entre todos los miembros del equipo durante todo el tiempo, cualquier resistencia del cliente o del equipo de desarrollo hacia las prácticas y principios puede llevar al proceso al fracaso (el clima de trabajo, la colaboración y la relación contractual son claves)
- El uso de tecnologías que no tengan un ciclo rápido de realimentación o que no soporten fácilmente el cambio, etc.

Aunque en la actualidad ya existen libros asociados a cada una de las metodologías ágiles existentes y también abundante información en Internet, es XP la metodología que resalta por contar con la mayor cantidad de información disponible.

Tomando en cuenta la información de la tabla 1.1 así como las necesidades del proyecto, se ha seleccionado la metodología XP (Extreme Programming) para ser aplicada en el proyecto.

Es correcto recalcar que debido al corto tiempo que se dispone para el desarrollo, no es necesario disponer de una metodología tradicional, además, el número de participantes en el grupo de trabajo es un factor que pesó bastante para la selección de la metodología XP.

1.3. PROTOCOLOS Y HERRAMIENTAS DE SOFTWARE Y HARDWARE

1.3.1. PROTOCOLOS

Aquí se ofrece una lista de los protocolos básicos que especifican la NFC:

- ISO / IEC 18092 o ECMA -340 (NFCIP - 1)

NFCIP significa Interfaz de comunicación de campo cercano y el protocolo. Se normaliza la capa física y la comunicación entre dos dispositivos NFC.

- ISO / IEC 21481 o ECMA 352 (NFCIP - 2)

Define un mecanismo de selección de modo de comunicación entre diferentes tecnologías sin contacto que operan en la frecuencia 13,56 Mhz.

- ISO / IEC 14443

La norma describe tarjetas de proximidad, métodos de operación y los protocolos de transmisión para la comunicación entre la tarjeta y el dispositivo lector. La tarjeta de proximidad es una tarjeta inteligente sin contacto con el intervalo aproximado de comunicación hasta 15 cm.

- ISO / IEC 15693

La norma describe un método de funcionamiento y operación de tarjetas inteligentes. Los soportes de datos en estas tarjetas suelen ser más baratos y ofrecer funciones más sencillas en el caso de las tarjetas de proximidad. (Finkenzeller, 2010)

1.3.1.1. ISO / IEC 14443¹¹

ISO 14443 es un estándar internacional relacionado con las tarjetas de identificación electrónicas, en especial las tarjetas inteligentes, gestionado conjuntamente por la Organización Internacional de Normalización (ISO) y la Comisión Electrónica Internacional (IEC).

Este estándar define una tarjeta de proximidad utilizada para identificación y pagos que por lo general utiliza el estándar tarjeta de crédito definida por ISO 7816 - ID 1.

El sistema RFID utiliza un lector con un micro controlador incrustado y una antena que opera a 13,56 MHz. El lector mantiene a su alrededor un campo electromagnético de modo que al acercarse una tarjeta al campo, ésta se alimenta eléctricamente de esta energía inducida y puede establecerse la comunicación lector-tarjeta. (Advanced Card Systems)

El estándar ISO 14443 consta de cuatro partes y se describen dos tipos de tarjetas: tipo A y tipo B. Las principales diferencias entre estos tipos se encuentran en los métodos de modulación, codificación de los planes (parte 2) y el protocolo de inicialización de los procedimientos (parte 3). Las tarjetas de ambos tipos (A y B) utilizan el mismo protocolo de alto nivel (llamado T=CL) que se describe en la parte 4. El protocolo T=CL especifica los bloques de datos y los mecanismos de intercambio:

- Bloque de datos de encadenamiento
- Tiempo de espera de extensión
- Múltiple activación

¹¹http://www.etsi.org/deliver/etsi_ts/102600_102699/10269501/08.01.00_60/ts_10269501v080100p.pdf

1.3.2. SOFTWARE

Dentro de este tema se va a tratar de enfocar sobre sistemas operativos móviles que han integrado en su core compatibilidad con la tecnología NFC basándose en estándares internacionales para su compatibilidad con los distintos tags y lectores existentes en el mercado.

1.3.2.1. Android

Android es un sistema operativo móvil diseñado principalmente para equipos móviles que cuenten con pantalla táctil.

Desde el año 2011, Android incorporo en su sistema la capacidad de operar con tags NFC, sin embargo, contaba con funciones muy limitadas ya que únicamente permitía leer la información de los tags, mas no manipularla o escribir nueva información en ellos.

La tecnología NFC no contó con gran apoyo en el 2011 por parte de los fabricantes de software, razón por la cual esta tecnología no fue conocida en los dispositivos móviles. Fue en el año 2013 cuando esta tecnología gana terreno al llamar la atención de fabricantes de celulares como Samsung, Sony y Motorola.

1.3.3. HARDWARE

1.3.3.1. TAG NFC

Son dispositivos que cuentan con memoria capaz de almacenar datos. En sí, son dispositivos que almacenan información que puede ser obtenida por lectores NFC.

Las etiquetas son un caso muy común de tags, sin embargo, los smartphones han logrado ser considerados como dispositivos NFC en los últimos años gracias a la incorporación de hardware y software dentro de los mismos.

Con el fin de cumplir con los objetivos de este nos enfocaremos en el análisis y estudio de smartphones que integren tecnología NFC en su sistema.

1.3.3.2. Lector NFC

Es un dispositivo capaz de captar los datos almacenados en tags NFC o en dispositivos inteligentes con tecnología NFC. Los lectores NFC pueden ser clasificados en dos grandes grupos, estáticos o móviles. Los lectores estáticos son aquellos que están posicionados en una única ubicación anclados a un PC u otros dispositivos para su funcionamiento mientras que los lectores NFC móviles son aquellos lectores que brindan la facilidad de ser transportados y operados durante el transporte.

Los teléfonos inteligentes entran dentro del grupo de lectores NFC móviles

1.4. ANÁLISIS DE DISPOSITIVOS INVOLUCRADOS

Componentes básicos de actuación en el mundo de la NFC son dispositivos con NFC y etiquetas NFC, también llamados etiquetas inteligentes. Estos suelen ser dispositivos pasivos (sin fuente de alimentación), que almacenan datos a ser leídos por el dispositivo con NFC, en principio, se trata de una etiqueta RFID compatible con la tecnología NFC. Actualmente son comúnmente conocidos como ContactLesscards (tarjetas sin contacto), ya que la tarjeta y la etiqueta son técnicamente los mismos objetos, con variaciones en el factor de forma. Las Tarjetas Contactless usadas en venta de entradas y los pagos hoy pueden incluir tecnología adicional para almacenar y asegurar los datos.

El dispositivo habilitado para NFC es un dispositivo activo que opera en uno de los modos NFC que se detallarán en posteriores capítulos y se utiliza para la comunicación con las etiquetas, tarjetas u otros dispositivos habilitados para NFC. Hoy en día, los dispositivos habilitados con NFC más ampliamente disponibles son los teléfonos móviles (especialmente smartphones). Además de los teléfonos móviles, otros dispositivos compatibles con NFC están disponibles. Ejemplos de ello son el punto de venta (POS) que pueden realizar los pagos sin contacto, diversos accesorios como Bluetooth audífonos, lentes de cámaras, dispositivos médicos, etc.

1.4.1. MOTO X (TAG NFC)¹²

El Motorola Moto X es el primer smartphone de Motorola bajo la supervisión de Google. El Moto X posee una pantalla AMOLED 720p de 4.7 pulgadas, procesador dual-core Snapdragon S4 Pro, 2GB de RAM, 16GB o 32GB de almacenamiento interno con Android 4.2.2 JellyBean. Incorpora la tecnología NFC dentro de su hardware.

Para utilizar la tecnología NFC en el teléfono, el teléfono debe estar dentro de aproximadamente 0,78 pulgadas (2 cm) de los teléfonos inteligentes con NFC, accesorio inteligente, punto de acceso con NFC o etiqueta inteligente con la que desee intercambiar información. (Motorola)

1.4.2. ACR 122 (LECTOR NFC)¹³

El ACR122U NFC Reader es un dispositivo de lectura y escritura ligado a un PC mediante una interfaz USB, desarrollado basándose en la tecnología RFID de 13.56 MHz. Compatible con el estándar ISO/IEC18092 para Near Field Communication (NFC), que apoya no sólo las tarjetas Mifare® e ISO 14443 A y B, sino también a los cuatro tipos de etiquetas NFC.

¹²<https://www.motorola.com/us/moto-x-specifications.html>

¹³<http://www.acs.com.hk/en/products/3/acr122u-usb-nfc-reader/>

ACR122U es compatible tanto con CCID y PC / SC. Por lo tanto, es un dispositivo plug- and-play USB que permite la interoperabilidad con diferentes dispositivos y aplicaciones. Con una velocidad de acceso de hasta 424 kbps y una velocidad total de USB de hasta 12 Mbps. La distancia de funcionamiento del ACR122U es de hasta 5 cm, dependiendo del tipo de contacto de la etiqueta en uso. (Advanced Card Systems)

Con el fin de aumentar el nivel de seguridad, ACR122U puede estar integrado con una ranura SAM ISO 7816-3. Además, el ACR122U NFC Reader está disponible en forma de módulo, permitiendo una fácil integración en máquinas más grandes, tales como terminales de punto de venta, sistemas de acceso físico y máquinas expendedoras.(Advanced Card Systems)

El ACR122U NFC Reader es ideal tanto para la verificación de identidad personal segura y transacciones de micro- pago en línea. Otras aplicaciones de la ACR122U incluyen control de acceso, los pagos electrónicos, e –ticketing para eventos y transporte público, recolección de tarifas de peaje de carreteras y la autenticación de red.

CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL PROTOTIPO

2.1. REQUERIMIENTOS

Se definirán los requerimientos mediante entrevistas realizadas a los usuarios potenciales del sistema.

2.1.1. REGISTRO DE EMPRESA

Descripción: el administrador tendrá la facultad de registrar empresas en el sistema.

Observaciones: para poder garantizar el éxito en el registro de empresas, será necesario proporcionar la siguiente información:

- Nombre comercial
- RUC
- Hora de ingreso - Hora de salida

2.1.2. ACTUALIZACIÓN DE EMPRESA

Descripción: el administrador podrá actualizar datos de las empresas registradas. Los siguientes datos podrán ser actualizados:

- Nombre Comercial
- Hora de ingreso - Hora de salida
- Habilitación –Deshabilitación de empresa

2.1.3. REGISTRO DE EMPLEADOS

Descripción: El administrador ingresará información de relevancia para identificar al personal de la organización

Observaciones: para poder garantizar el éxito en el registro de usuarios, será necesario proporcionar la siguiente información:

- Nombres - Apellidos
- Cédula de ciudadanía o Número de pasaporte
- PIN de celular
- Empresa

Se podrá ingresar información adicional no obligatoria como:

- Fecha de caducidad de contrato laboral
- Hora de ingreso - Hora de salida

2.1.4. ACTUALIZACIÓN DE EMPLEADOS

Descripción: el administrador podrá actualizar datos del personal.

Los siguientes datos podrán ser actualizados:

- Fecha de caducidad de contrato laboral
- Hora de ingreso - Hora de salida
- PIN celular
- Habilitación –Deshabilitación de usuario

2.1.5. MONITOREAR ACCESOS

Descripción: permite visualizar los registros tomados por el lector NFC en tiempo real

2.1.6. REGISTRO DE ACCESOS

Descripción: el administrador podrá almacenar información de los accesos a los cuales se asignarán un lector NFC.

Observaciones: para poder garantizar el éxito en el registro de accesos, será necesario proporcionar la siguiente información:

- Código de acceso
- Descripción de acceso

2.1.7. ACTUALIZACIÓN DE ACCESOS

Descripción: el administrador podrá actualizar datos de los accesos. Los siguientes datos podrán ser actualizados:

- Código de acceso
- Descripción de acceso

2.1.8. REGISTRO DE ÁREAS

Descripción: el administrador podrá registrar las áreas o zonas de la organización en la cual se implantarán controles de acceso.

Observaciones: para poder garantizar el éxito en el registro de áreas, será necesario proporcionar la siguiente información:

- Nombre del área
- Código de área
- Lector asignado al área

Se podrá ingresar información adicional no obligatoria como:

- Descripción de área

2.1.9. ACTUALIZACIÓN DE ÁREA

Descripción: el administrador podrá actualizar datos de las áreas. Los siguientes datos podrán ser actualizados:

- Nombre de área
- Descripción de área
- Lector asignado al área

2.1.10. REGISTRO DE LECTORES

Descripción: el administrador podrá registrar lectores NFC que se interesen incorporar al sistema de control de acceso.

Observaciones: para poder garantizar el éxito en el registro de lectores, será necesario proporcionar la siguiente información:

- Modelo del lector
- Identificador del Lector
- Fabricante del lector

Se podrá ingresar información adicional no obligatoria como:

- Descripción del lector

El modulo dispondrá de la siguiente funcionalidad añadida

- Detección automática de lectores conectados

2.1.11. ACTUALIZACIÓN DE LECTORES

Descripción: el administrador podrá actualizar datos de los lectores registrados en el sistema. Los siguientes datos podrán ser actualizados:

- Descripción

2.2. HISTORIAS DE USUARIO

Cada historia de usuario debe de contener los siguientes elementos:

- Código (número)
- Nombre (funcionalidad)
- Prioridad
- Descripción
- Riesgo (desarrollo)
- Observaciones (Puntos a considerar)

2.2.1. REGISTRO DE EMPRESAS

Número: 01	Prioridad: alta	Riesgo: bajo
Nombre: Registro de empresas		
Descripción: El administrador del sistema necesita registrar empresas para poder agrupar a los empleados y accesos dentro de esta.		
Observaciones: Es de importancia registrar la siguiente información: Nombre comercial de la empresa		

RUC de la empresa Hora de ingreso de los empleados Hora de salida de los empleados
--

Tabla 2.1 Historia de usuario: Registro de empresas

Criterios de aceptación:

- El usuario quiere poder registrar la empresa guardando los datos más relevantes.
- El usuario no debe poder registrar dos empresas con un mismo RUC
- El usuario no debe poder registrar una empresa sin el número de RUC y la razón social.

2.2.2. ACTUALIZACIÓN DE EMPRESAS

Número: 02	Prioridad: baja	Riesgo: bajo
Nombre: Actualización de empresas		
Descripción: El administrador del sistema necesita actualizar datos de la empresa para poder corregir errores que se pudieron haber generado.		
Observaciones: Se requiere actualizar los siguientes datos Hora de ingreso Hora de salida Nombre comercial de la empresa		

Tabla 2.2 Historia de usuario: Actualización de empresas

Criterios de aceptación:

- El usuario no debe poder actualizar el RUC de la empresa.
- El usuario podrá únicamente actualizar los datos informativos de la empresa, no los identificativos.
- El usuario desea ser notificado mediante una alerta que la actualización se ha realizado exitosamente.

2.2.3. CAMBIO DE ESTADO

Número: 03	Prioridad: media	Riesgo: alto
Nombre: Cambio de estado de la empresa		
Descripción: El administrador del sistema requiere cambiar de estado de la empresa para poder controlar las actividades de los accesos en una empresa.		
Observaciones: Si una empresa es creada, la empresa nacerá con estado "VIGENTE". Este estado puede ser cambiado a "NO VIGENTE" en caso de ser requerido.		

Tabla 2.3 Historia de usuario: Cambio de estado de empresa

Criterios de aceptación:

- El usuario debe de ser notificado mediante una alerta visual que el actualización de estado se ha realizado exitosamente.

- El usuario quiere diferenciar visualmente el estado de una empresa, esto puede ser mediante una coloración distinta o un icono distintivo.

2.2.4. REGISTRO DE EMPLEADOS

Número: 04	Prioridad: alta	Riesgo: bajo
Nombre: Registro de empleados		
Descripción: El administrador del sistema necesita registrar cada uno de los empleados de la empresa para poder controlar los accesos a la empresa.		
Observaciones: Únicamente es necesario registrar los empleados que vayan a emplear el control de acceso en la organización. Es necesario almacenar los siguientes datos: Nombre del empleado Apellido del empleado Identificación del empleado PIN del empleado (pin celular NFC) Empresa Hora de ingreso Hora de salida		

Tabla 2.4 Historia de usuario: Registro de empleados

Criterios de aceptación:

- El usuario quiere registrar en el sistema datos relevantes del empleado.
- El usuario no debe poder registrar un empleado con el mismo identificativo.
- El usuario debe registrar obligatoriamente el PIN.
- El usuario quiere tener una alerta visual que indique si el empleado ha sido registrado exitosamente.
- El usuario quiere tener una alerta visual que le indique las razones por las cuales un empleado no pudo ser registrado.

2.2.5. CAMBIO DE ESTADO DE EMPLEADO

Número: 05	Prioridad: baja	Riesgo: alta
Nombre: Cambio de estado del empleado		
Descripción: El administrador del sistema necesita cambiar el estado de un empleado registrado en el sistema para poder controlar la vigencia de un empleado en la empresa.		
Observaciones: Los empleados registrados en el sistema pueden estar habilitados o deshabilitados. Un empleado no habilitado generará una alerta en el sistema en caso de que sea leído por un lector NFC		

Tabla 2.5 Historia de usuario: Cambio de estado de empleado

Criterios de aceptación:

- El usuario debe de ser notificado mediante una alerta visual que el actualización de estado se ha realizado exitosamente.

- El usuario quiere diferenciar visualmente el estado de un empleado, esto puede ser mediante una coloración distinta o un icono distintivo.

2.2.6. ACTUALIZACIÓN DE EMPLEADO

Número: 06	Prioridad: baja	Riesgo: bajo
Nombre: Actualización de datos del empleado		
Descripción: El administrador del sistema requiere actualizar los datos de un empleado registrado en el sistema para poder mejorar la calidad de los datos.		
Observaciones: Las actualizaciones de datos no serán aplicadas para el nombre, el apellido y el id del empleado		

Tabla 2.6 Historia de usuario: Actualización de datos del empleado

Criterios de aceptación:

- El usuario quiere tener una alerta visual que indique si el empleado ha sido actualizado exitosamente.
- El usuario quiere tener una alerta visual que le indique las razones por las cuales un empleado no pudo ser actualizado.
- El usuario únicamente podrá actualizar los datos informativos del empleado, no los datos identificativos.
- El usuario desea visualizar la fecha en la cual se realizó la última actualización de un empleado.

2.2.7. REGISTRO DE LECTOR

Número: 07	Prioridad: alta	Riesgo: alta
Nombre: Registro de lector		
Descripción: El administrador del sistema requiere registrar los lectores que la empresa decida tener para controlar las distintas áreas.		
Observaciones: Es necesario registrar la siguiente información: Modelo del lector Identificativo del lector Fabricante del lector Descripción del lector		

Tabla 2.7 Historia de usuario: Registro de lector

Criterios de aceptación:

- El usuario quiere registrar en el sistema datos relevantes de los lectores NFC.
- El usuario no debe poder registrar un lector con el mismo identificativo.
- El usuario quiere tener una alerta visual que indique si el lector ha sido registrado exitosamente.

- El usuario quiere tener una alerta visual que le indique las razones por las cuales un lector NFC no pudo ser registrado.

2.2.8. ACTUALIZACIÓN DE LECTOR

Número: 08	Prioridad: alta	Riesgo: alta
Nombre: Actualización de lectores		
Descripción: El administrador del sistema requiere actualizar datos de los lectores registrados en el sistema para poder mejorar la calidad de los datos.		
Observaciones: Es necesario actualizar los siguientes datos: Descripción del lector		

[Tabla 2.8 Historia de usuario: Actualización de datos del lector](#)

Criterios de aceptación:

- El usuario quiere tener una alerta visual que indique si el lector ha sido actualizado exitosamente.
- El usuario quiere tener una alerta visual que le indique las razones por las cuales el lector no pudo ser actualizado.
- El usuario únicamente podrá actualizar los datos informativos del lector, no los datos identificativos.
- El usuario desea visualizar la fecha en la cual se realizó la última actualización de un lector.

2.2.9. ELIMINAR LECTOR

Número: 09	Prioridad: baja	Riesgo: alta
Nombre: Eliminar lector del sistema		
Descripción: El administrador del sistema requiere eliminar lectores previamente registrados en el sistema para poder depurar lectores sin uso.		
Observaciones: Para eliminar un lector será necesario desvincular los lectores de las áreas creadas en el sistema		

[Tabla 2.9 Historia de usuario: Eliminar lector](#)

Criterios de aceptación:

- El usuario debe de tener la posibilidad de confirmar la eliminación de un lector NFC mediante un cuadro de diálogo.

2.2.10. RESGISTRAR ACCESO

Número: 10	Prioridad: alta	Riesgo: alta
Nombre: Registro de accesos (área)		
Descripción: El administrador del sistema requiere registrar los accesos o áreas de la empresa que se deseen controlar.		

<p>Observaciones: Es necesario registrar la siguiente información: Nombre del área Código de área Descripción del área Lector asignado para el área</p>

Tabla 2.10 Historia de usuario: Registro de accesos

Criterios de aceptación:

- El usuario quiere registrar en el sistema datos relevantes de los accesos.
- El usuario no debe poder registrar un acceso con el mismo identificador.
- El usuario quiere tener una alerta visual que indique si el acceso ha sido registrado exitosamente.
- El usuario quiere tener una alerta visual que le indique las razones por las cuales un acceso no pudo ser registrado.

2.2.11. ACTUALIZACIÓN DE ACCESOS

Número: 11	Prioridad: baja	Riesgo: baja
Nombre: Actualización de accesos		
Descripción: El administrador del sistema requiere actualizar los datos de los accesos o áreas de la empresa registradas en el sistema.		
Observaciones: Los datos que podrán ser actualizados son los siguientes: Nombre del área Descripción del área Lector asignado para el área		

Tabla 2.11 Historia de usuario: Actualización de accesos

Criterios de aceptación:

- El usuario quiere tener una alerta visual que indique si el acceso ha sido actualizado exitosamente.
- El usuario quiere tener una alerta visual que le indique las razones por las cuales el acceso no pudo ser actualizado.
- El usuario únicamente podrá actualizar los datos informativos de los accesos, no los datos identificativos.
- El usuario desea visualizar la fecha en la cual se realizó la última actualización de un acceso.

2.2.12. ELIMINACIÓN DE ACCESOS

Número: 12	Prioridad: baja	Riesgo: Alto
Nombre: Eliminación de accesos		
Descripción: El administrador del sistema requiere eliminar los datos de los accesos o áreas de la empresa registradas en el sistema.		
Observaciones:		

La eliminación de un acceso del sistema implica la desvinculación de todos los datos relacionados con esta área o acceso

Tabla 2.12 Historia de usuario: Eliminación de accesos

Criterios de aceptación:

- El usuario debe de tener la posibilidad de confirmar la eliminación de un acceso mediante un cuadro de diálogo.

2.2.13. ENLACE CON LECTORES

Número: 13	Prioridad: alta	Riesgo: alta
Nombre: Enlace con lector NFC		
Descripción: El sistema requiere que se enlace con lectores NFC para poder obtener los datos de estos lectores.		
Observaciones: Es necesario trabajar con un lector NFC específico ya que se requiere un desarrollo único para cada driver empleado por cada lector.		

Tabla 2.13 Historia de usuario: Enlace con lector NFC

Criterios de aceptación:

- El sistema debe de enlazarse con el modelo de lector ACR122U.
- El usuario debe poder visualizar mediante una alerta si existe algún inconveniente en el enlace.

2.2.14. CAPTURA DE DATOS

Número: 14	Prioridad: alta	Riesgo: alta
Nombre: Captura de datos desde lector NFC		
Descripción: Se requiere obtener los datos que capture el lector NFC y almacenarlos para un análisis posterior.		
Observaciones: Todos los datos obtenidos desde los lectores serán almacenados en una base de datos. El administrador del sistema podrá controlar el inicio de la captura de datos de los lectores de acuerdo a las necesidades y criterios establecidos por la empresa.		

Tabla 2.14 Historia de usuario: Captura de datos desde el lector NFC

Criterios de aceptación:

- Los datos deben de ser capturados de manera automática.
- Se debe de llenar un log de los datos que se capturen.
- El usuario requiere guardar el log en disco para su posterior análisis.

2.3. PLAN DE ENTREGAS

Una vez definidas las funcionalidades que van a ser desarrolladas por el sistema, es importante realizar un plan de entregas. Este plan de entregas es el resultado de una reunión entre todos los actores del proyecto (cliente,

desarrolladores, gerentes, etc.). En XP este proceso es conocido comúnmente como el juego de planificación (Planning Game).

El cliente ordena y agrupa según sus prioridades las historias de usuario. El cronograma de entregas se realiza en base a las estimaciones de tiempos de desarrollo realizadas por los desarrolladores.

Para poder realizar la estimación de tiempo y esfuerzo se tomarán en cuenta los siguientes aspectos:

- La jornada de trabajo será de 8 horas.
- El equipo de desarrollo está conformado por dos personas.
- La prioridad de las historias de usuario para la empresa.
- El riesgo del desarrollo de cada historia de usuario.

En la tabla 2.15 se muestra la estimación de esfuerzos para el desarrollo del sistema.

Nº	Nombre de historia	Riesgo	Prioridad	Tiempo (días)
1	Registro de empresas	Bajo	Alta	4
2	Actualización de empresas	Bajo	Baja	4
3	Cambio de estado de la empresa	Medio	Alta	3
4	Registro de empleados	Alto	Baja	5
5	Cambio de estado del empleado	Bajo	Baja	3
6	Actualización de datos del empleado	Bajo	Baja	3
7	Registro de lector	Alto	Alto	8
8	Actualización de lectores	Alto	Alta	5
9	Eliminar lector del sistema	Alto	Baja	5
10	Registro de acceso	Alto	Alto	7
11	Actualización de accesos	Bajo	Baja	4
12	Eliminación de accesos	Alto	Baja	4
13	Enlace con lector NFC	Alto	Alto	15
14	Captura de datos desde el lector	Alto	Alto	15

Tabla 2.15: Estimación de esfuerzos

2.4. PRIORIZACIÓN

Basándonos en la estimación de esfuerzo, tenemos el tiempo que tomará el desarrollo de cada historia de usuario. La priorización del desarrollo de estas historias de usuario estará determinada por la importancia que tenga esa funcionalidad para la empresa y la complejidad en su desarrollo, además del riesgo que pueda presentarse por diversos factores en el desarrollo. Con estos datos podemos establecer el orden de entrega de las historias de usuario, designando de igual manera a una iteración tentativa. En la tabla 2.16 se indica la iteración en la cual va a ser desarrollada cada historia de usuario.

Nº	Nombre de historia	Iteración
1	Registro de empresas	1
2	Actualización de empresas	3
3	Cambio de estado de la empresa	3
4	Registro de empleados	1
5	Cambio de estado del empleado	3
6	Actualización de datos del empleado	3
7	Registro de lector	1
8	Actualización de lectores	3
9	Eliminar lector del sistema	3
10	Registro de acceso	1
11	Actualización de accesos	3
12	Eliminación de accesos	3
13	Enlace con lector NFC	2
14	Captura de datos desde el lector	2

Tabla 2.16: Priorización de historias de usuario

2.5. PLAN DE ENTREGAS

En cada iteración se establecen fechas para el desarrollo de cada funcionalidad. En la tabla 2.17 podemos observar la historia de usuario, el inicio de la tarea, el fin de la tarea y la iteración a la cual pertenece dicha historia.

Nº	Nombre de historia	Iteración	Fecha inicio	Fecha fin
1	Registro de empresas	1	05/01/2014	20/03/2014
2	Actualización de empresas	3	02/06/2014	30/07/2014
3	Cambio de estado de la empresa	3	02/06/2014	30/07/2014
4	Registro de empleados	1	05/01/2014	20/03/2014
5	Cambio de estado del empleado	3	02/06/2014	30/07/2014
6	Actualización de datos del empleado	3	02/06/2014	30/07/2014
7	Registro de lector	1	05/01/2014	20/03/2014
8	Actualización de lectores	3	02/06/2014	30/07/2014
9	Eliminar lector del sistema	3	02/06/2014	30/07/2014
10	Registro de acceso	1	05/01/2014	20/03/2014
11	Actualización de accesos	3	02/06/2014	30/07/2014
12	Eliminación de accesos	3	02/06/2014	30/07/2014
13	Enlace con lector NFC	2	25/03/2014	30/05/2014
14	Captura de datos desde el lector	2	25/03/2014	30/05/2014

Tabla 2.17: Plan de entregas

2.6. TARJETAS CRC

La técnica CRC propone una forma de trabajo, preferentemente grupal, para encontrar los objetos del dominio de la aplicación, sus responsabilidades y

cómo colaboran con otros para realizar tareas. Esta técnica utiliza las llamadas tarjetas CRC, las cuáles registran el nombre de las clases, sus responsabilidades y las otras clases con la que colaboran. Además de la forma de registrar la información de las clases, la técnica propone un proceso llamado proceso CRC.

Las principales características de las tarjetas son:

- Identificación de clases y asociaciones que participan del diseño del sistema.
- Obtención de las responsabilidades que debe cumplir cada clase.
- Establecimiento de cómo una clase colabora con otras clases para cumplir con sus responsabilidades.

2.6.1. EMPLEADO

EMPLEADO	
RESPONSABILIDADES	COLABORADORES
<ul style="list-style-type: none"> • Registrar empleados • Actualizar datos de empleados • Consultar existencia de empleados 	<ul style="list-style-type: none"> • Empresa

Tabla 2.18 CRC: Empleado

2.6.2. EMPRESA

EMPRESA	
RESPONSABILIDADES	COLABORADORES
<ul style="list-style-type: none"> • Registro de empresa • Actualizar datos de la empresa • Consultar datos existentes de empresa 	

Tabla 2.19 CRC: Empresa

2.6.3. ÁREA

ÁREA	
RESPONSABILIDADES	COLABORADORES
<ul style="list-style-type: none"> • Registrar área • Actualizar datos del área registrada • Consultar datos del área registrada 	<ul style="list-style-type: none"> • empresa • lector

Tabla 2.20 CRC: Área

2.6.4. LECTOR

LECTOR	
RESPONSABILIDADES	COLABORADORES
<ul style="list-style-type: none"> • Registrar lector • Actualizar datos de lector registrado • Eliminar lector registrado • Buscar lectores 	<ul style="list-style-type: none"> • empresa

- | |
|--|
| registrados <ul style="list-style-type: none"> • Buscar lectores conectados en la red |
|--|

Tabla 2.21 CRC: Lector

2.6.5. MONITOR

MONITOR	
RESPONSABILIDADES	COLABORADORES
<ul style="list-style-type: none"> • Buscar lectores conectados a la red • Buscar lectores registrados en el sistema • Iniciar proceso de lectura continua de tags • Capturar datos de tags NFC • Almacenar datos capturados 	<ul style="list-style-type: none"> • lector • empresa • área • empleado

Tabla 2.22 CRC: Monitor

2.6.6. TAG

TAG NFC	
RESPONSABILIDADES	COLABORADORES
<ul style="list-style-type: none"> • Entregar datos almacenados • Establecer conexión con lector NFC 	<ul style="list-style-type: none"> • empleado • empresa

Tabla 2.23 CRC: TAG

2.7. ANÁLISIS

Pressman establece que la tarea del análisis de requisitos es un proceso de descubrimiento, refinamiento, modelado y especificación¹⁴.

El análisis de requisitos permite al desarrollador o desarrolladores especificar la función y el rendimiento del software, indica la interfaz del software con otros elementos del sistema y establece las restricciones que debe cumplir el software. Puede dividirse en cinco áreas de esfuerzo que son:

- Reconocimiento del problema
- Evaluación y síntesis
- Modelado
- Especificación
- Revisión

¹⁴http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/fuentes_k_jf/capitulo2.pdf

2.7.1. SITUACIÓN ACTUAL

En la actualidad, se ha visto la necesidad de implementar políticas que permitan controlar y garantizar el cumplimiento de normas establecidas por los altos directivos de las organizaciones.

El Control de Acceso es uno de estos controles el cual consiste en un componente de seguridad que monitorea automáticamente y controla el tráfico a través de entornos físicos como, puertas, entradas y ascensores.

Identificando y registrando los accesos a las instalaciones ayuda a llevar un control adecuado y real del entorno de una organización. Oficinas, fábricas, escuelas, clubes, etc., pueden mantener un eficiente control de los accesos.

Registrar horarios de entradas y salidas del personal se ha convertido en una tarea esencial para toda organización en crecimiento, es por esta razón que se tomó la decisión de realizar un desarrollo adecuado para el registro de ingresos y salidas del personal de Mivsell.

En la actualidad la empresa no cuenta con un sistema de control de accesos, sin embargo, dispone de teléfonos celulares Android que incorporan tecnología NFC, razón por la cual se optó por emplear estos dispositivos como lectores de tarjetas inteligentes.

2.8. DISEÑO

2.8.1. ENTORNO DE DESARROLLO

2.8.1.1. Herramientas CASE

Herramientas CASE (ComputerAided Software Engineering, Ingeniería de Software Asistida por Computadoras). Son diversas Aplicaciones informáticas destinadas a aumentar la productividad en el Desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas ayudan en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el diseño de proyectos, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, Compilación automática, documentación o detección de errores entre otras.

De acuerdo con Kendall y Kendall "la ingeniería de sistemas asistida por ordenador es la aplicación de tecnología informática a las actividades, las técnicas y las metodologías propias de desarrollo, su objetivo es acelerar el proceso para el que han sido diseñadas, en el caso de CASE para automatizar o apoyar una o más fases del ciclo de vida del desarrollo de sistemas"¹⁵.

¹⁵<http://mundoinformatico321.blogspot.com/2012/11/metodologia-kendall-kendall.html>

Cuando se hace la planificación de la base de datos, la primera etapa del ciclo de vida de las aplicaciones de bases de datos, también se puede escoger una herramienta CASE que permita llevar a cabo el resto de tareas del modo más eficiente y efectivo posible. Una herramienta CASE suele incluir:

- Un diccionario de datos para almacenar información sobre los datos de la aplicación de bases de datos.
- Herramientas de diseño para dar apoyo al análisis de datos.
- Herramientas que permitan desarrollar el modelo de datos corporativo, así como los esquemas conceptual y lógico.
- Herramientas para desarrollar los prototipos de las aplicaciones.

RAD propone emplear herramientas existentes en la actualidad para agilizar el desarrollo de un producto de Software, basándose en criterios de selección que vayan acorde con los requerimientos del proyecto.

2.8.1.2. Criterios de selección de herramientas CASE

Analizando el alcance del proyecto y los requerimientos del sistema, se creará una lista de herramientas CASE a ser utilizadas durante el desarrollo del proyecto, basando en ciertos criterios que se detallan a continuación:

1. Las herramientas a seleccionarse deben ser capaces de resolver una amplia gama de problemas de integración de aplicaciones empresariales.
2. Las herramientas a seleccionarse deben ser capaces de proporcionar una comunicación con un entorno JAVA.
3. Las herramientas a seleccionarse deben ser licenciadas bajo esquemas de licenciamiento libre.
4. Las herramientas deben estar en versiones estables.
5. Las herramientas deben proporcionar una comunicación con la base de datos PostgreSQL V9.3.
6. Las herramientas deben garantizar agilidad en el desarrollo de las interfaces de usuario necesarias para cumplir con los requerimientos del software.
7. Las herramientas den ayudar al diseño de un modelo de datos.

Basándose en los criterios mencionados, se han seleccionado las siguientes herramientas:

Tabla 2.24 Herramientas CASE

Nombre	Versión	Desarrollador	Descripción
SceneBuilder	2.0	Oracle Corporation	Herramienta que permite agilizar el diseño de interfaces
DBVisualizer	9.1	Oracle Corporation	Herramienta universal de modelamiento de

			datos
PgAdmin	1.1	PgAdmindevelopmentTeam	Herramienta para administración de base de datos

2.8.2. ARQUITECTURA

Para asegurar que el desarrollo de una aplicación satisface las necesidades de los usuarios, se modelará una arquitectura la cual permita ayudar a reducir las ambigüedades que se puedan dar por las explicaciones en el lenguaje natural.

2.8.2.1. Arquitectura Android - NFC

Con el modo de emulación de tarjeta, los datos son enviados al CPU en el que las aplicaciones de Android se ejecutan directamente, en lugar de encaminar las tramas del protocolo NFC a un elemento seguro, estos datos son procesados gracias al controlador NFC que incorpora el smartphone empleado para el proyecto.

En la figura 2.1 se detalla la infraestructura empleada para emplear la tecnología NFC con plataformas Android

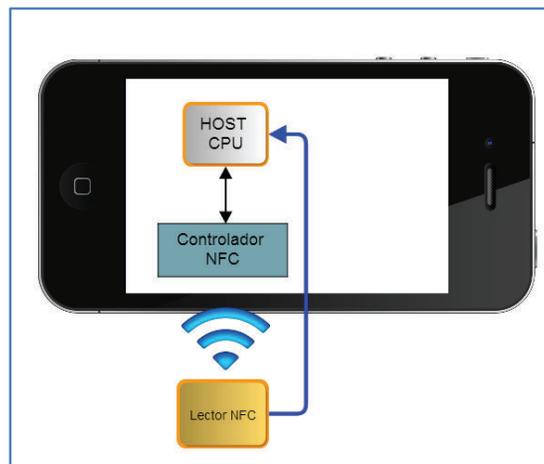


Figura 2.1 Arquitectura NFC

Con HCE, cualquier aplicación en un dispositivo Android 4.4 puede emular una tarjeta inteligente de NFC, permitiendo a los usuarios aprovechar para iniciar operaciones con una aplicación de su elección.

La arquitectura a ser utilizada para el proyecto consiste en una arquitectura cliente - servidor. La arquitectura cliente - servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes. En este modelo las aplicaciones se dividen de forma que el servidor contiene la

parte que debe ser compartida por varios usuarios, y en el cliente permanece sólo lo particular de cada usuario. (CNXSoft)

En la figura 2.2 se detalla un esquema que ilustra la arquitectura a ser empleada en el desarrollo del proyecto

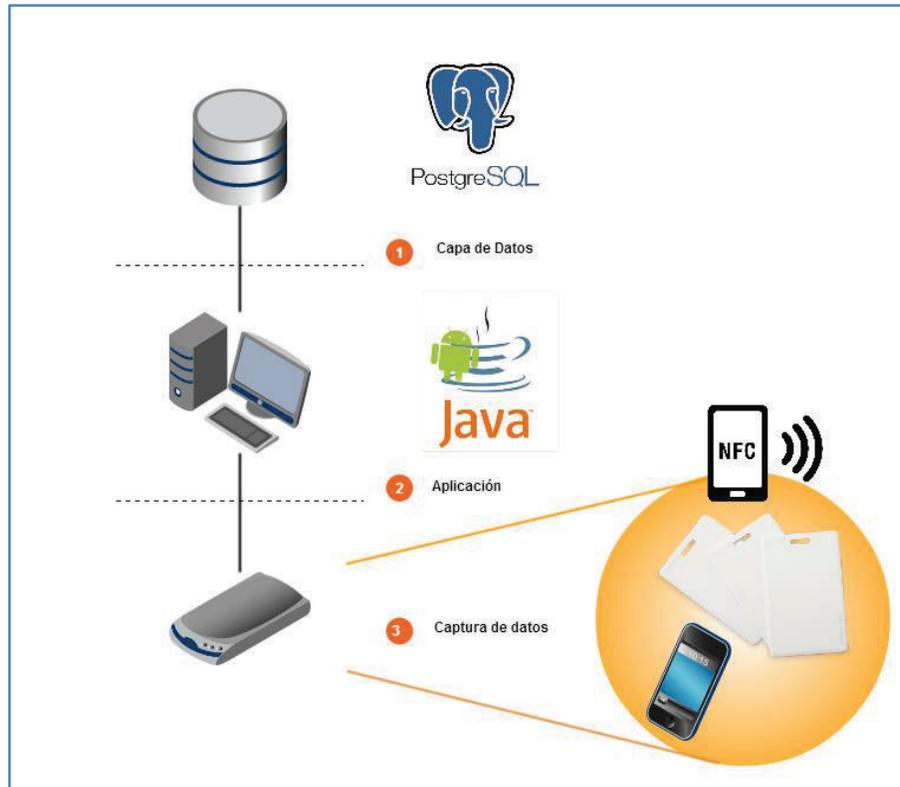


Figura 2.2 Arquitectura del sistema

Cada capa realiza las siguientes funciones

Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos.

Aplicación: presenta el sistema al usuario, comunica la información y captura la información del usuario en un mínimo de proceso en la cual se realiza un filtrado previo para comprobar que no existan errores de formato. También es conocida como interfaz gráfica y debe tener la característica de ser amigable, entendible y fácil de usar para el usuario.

Captura de datos: es donde se van a capturar los datos a través de un periférico el cual permite la comunicación entre distintos dispositivos.

Para el caso del proyecto en desarrollo se cuenta con un lector que permitirá recopilar los datos de acceso del personal obtenidos empleando la tecnología NFC.

2.8.3. DISEÑO DE INTERFACES

Con el fin de garantizar la acogida por los usuarios, se desarrollaran las interfaces tomando en cuenta ciertos consejos de relevancia en el diseño rápido de aplicaciones.

Estos son algunos de los consejos más importantes que se deben tener en cuenta a la hora de crear interfaces de manera usable:

- Hablar el lenguaje de los usuarios
 - Usar frases y conceptos familiares para el usuario.
 - Presentar la información en un orden natural y lógico para el usuario.
- Ser consistente
 - Indicar conceptos similares con igual terminología y gráfica.
 - Utilizar convenciones uniformes para la diagramación y el formato de la información.
- Minimizar la carga cognitiva
 - Tratar de utilizar el reconocimiento más que la memoria.
 - No forzar a los usuarios a recordar información entre documentos.
 - Otorgar enlaces directos a la información necesaria.
- Armar sistemas eficientes y flexibles
 - Diseñar un sistema que pueda ser utilizado por un rango amplio de usuarios.
 - Brindar instrucciones cuando sean necesarias para nuevos usuarios pero que no se entrometa en el camino de usuarios avanzados.
 - Permitir a los usuarios avanzados ir directamente al contenido que buscan.
- Diseñar sistemas con criterio minimalista y estético
 - Eliminar información irrelevante o déjela disponible sólo por pedido.
 - Tener en cuenta el diseño de factores humanos (ergonomía y diseño visual).
 - Tratar de generar una sensación positiva a nivel estético.
- Agrupación de la información
 - Tratar de ordenar la información para que los documentos sean auto contenidos
 - Cuando los documentos se hagan demasiados largos, proveer navegadores internos.
 - No forzar a los usuarios a acceder a múltiples documentos para tomar un sólo concepto.
- Proveer niveles de detalle progresivos
 - Organizar la información jerárquicamente, con la general antes que la particular.

- Aprovechar las características del hipertexto para proveer detalle en demanda, pero no incluirlo en donde hay información más general.

2.8.3.1. Registro de empresa

Descripción: el administrador tendrá la facultad de registrar empresas en el sistema.

En la figura 2.4 se muestra la interfaz del módulo de empresas la cual contiene los siguientes campos:

Nombre comercial: permite ingresar un nombre identificativo de la empresa

RUC: permite ingresar el número de registro único de contribuyente de la empresa.

Hora de ingreso: permite ingresar la hora de ingreso del personal según la constitución de la empresa.

Hora de salida: permite ingresar la hora de salida del personal según la constitución de la empresa.

	Nombre Empresa	RUC	Hora Ingreso	Hora Salida
Tabla sin contenido				

Nombre comercial

RUC

Hora de ingreso ⌚

Hora de salida ⌚

Figura 2.4 Registro de empresas

2.8.3.2. Actualización de empresa

Descripción: el administrador podrá actualizar datos de las empresas registradas. Los siguientes datos podrán ser actualizados:

- Nombre Comercial
- Hora de ingreso - Hora de salida
- Habilitación - Deshabilitación de empresa

2.8.3.3. Registro de empleados

Descripción: El administrador ingresará información de relevancia para identificar al personal de la organización.

En la figura 2.5 se muestra la interfaz del módulo de empleados la cual contiene los siguientes campos:

Nombre: permite ingresar el nombre o los nombres de los empleados.

Apellido: permite ingresar el apellido o los apellidos de los empleados.

Ingreso: hora en la cual el empleado debe de ingresar a la compañía. Esta hora por recomendación debe ser la hora de ingreso especificada en el contrato de trabajo del empleado.

Ingreso: hora en la cual el empleado debe de salir de la compañía. Esta hora por recomendación debe ser la hora de salida especificada en el contrato de trabajo del empleado.

ID: secuencia de caracteres que va a identificar al empleado.

PIN: identificativo obtenido del teléfono inteligente. Esta secuencia de caracteres relaciona al teléfono inteligente con NFC con el empleado.

Empresa: permite seleccionar la empresa a la cual pertenece el empleado.

Habilitado (Check box): permite habilitar al empleado la identificación a través del teléfono inteligente.

Leer (Botón): permite capturar el id del teléfono empleando un lector instalado en la PC donde se ejecute el registro del empleado.

Figura 2.5 Registro de empleados

2.8.3.4. Actualización de empleados

Descripción: el administrador podrá actualizar datos del personal. Los siguientes datos podrán ser actualizados:

- Fecha de caducidad de contrato laboral
- Hora de ingreso - Hora de salida
- PIN celular
- Habilitación - Deshabilitación de usuario

2.8.3.5. Monitorear accesos

Descripción: permite visualizar los registros tomados por el lector NFC en tiempo real

2.8.3.6. Registro de accesos

Descripción: el administrador podrá almacenar información de los accesos a los cuales se asignarán un lector NFC.

Observaciones: para poder garantizar el éxito en el registro de accesos, será necesario proporcionar la siguiente información:

- Código de acceso
- Descripción de acceso

2.8.3.7. Actualización de accesos

Descripción: el administrador podrá actualizar datos de los accesos. Los siguientes datos podrán ser actualizados:

- Código de acceso

2.8.3.8. Registro de áreas

Descripción: el administrador podrá registrar las áreas o zonas de la organización en la cual se implantarán controles de acceso.

En la figura 2.6 se muestra la interfaz del módulo de empleados la cual contiene los siguientes campos:

Nombre de área: nombre que permite describir a una área dentro de la empresa.

Código de área: permite ingresar una secuencia de caracteres que permite identificar a una área dentro de la empresa. Esta identificación debe de ser única para cada área.

Descripción: permite registrar datos de importancia que permitan identificar a una área de la empresa.

Lector: permite seleccionar un lector previamente registrado en el sistema.

The image shows a web application interface for managing areas. At the top, there is a header with the title "Areas". Below the header is a table with two columns labeled "C1" and "C2". The table is currently empty and displays the message "Tabla sin contenido". Below the table, there are three form fields: "Nombre de área", "Código de área", and "Descripción de área". The "Nombre de área" field is followed by a dropdown menu with "Lector" selected. To the right of the form fields, there are two buttons: "Guardar" and "Cancelar".

Figura 2.6

2.8.3.9. Registro de lectores

Descripción: el administrador podrá registrar lectores NFC que se interesen incorporar al sistema de control de acceso.

En la figura 2.7 se muestra la interfaz del módulo de lectores la cual contiene los siguientes campos:

Modelo: permite registrar el modelo del lector a ser empleado en la empresa para la captura de información de los teléfonos inteligentes con NFC.

ID: permite registrar el identificador del lector NFC.

Fabricante: permite registrar el fabricante del lector NFC

Descripción: permite registrar información relevante que permita identificar atributos del lector NFC.

Buscar lectores (botón): permite realizar la búsqueda de un lector NFC que esté conectado a la PC en la cual se pretende realizar el registro. Para poder

realizar la búsqueda del lector, se debe tener previamente instalado el driver provisto por el fabricante del lector en la PC.

The screenshot shows a window titled "Lectores". At the top, there is a table with the following headers: "Modelo", "ID", "Fabricante", and "Descripción". The table body is empty and contains the text "Tabla sin contenido". Below the table, there are four input fields labeled "Modelo", "ID", "Fabricante", and "Descripción". To the right of the "Modelo" field is a "Guardar" button. To the right of the "ID" field is a "Cancelar" button. To the right of the "Fabricante" field is a "Buscar Lectores" button.

Figura 2.7

2.8.3.10. Barra de acceso a funcionalidades

Descripción: permite el acceso a los módulos del sistema.

En la figura 2.8 se aprecia el diseño del menú de acceso a funcionalidades. Sobre este menú se aprecian la siguientes secciones:

- empresa
- usuarios
- Áreas
- Lectores
- Accesos
- Monitoreo



Figura 2.8

2.8.3.11. Barra de notificaciones

Descripción: Muestra información relacionada con el equipo en el cual se ejecuta el aplicativo.

En la figura 2.9 podemos visualizar la barra de notificaciones, sobre esta se despliega la siguiente información:

- Sesión de usuario: captura la sesión del sistema operativo sobre la cual se está ejecutando el aplicativo.
- Nombre del equipo: captura el nombre del PC sobre el cual se está ejecutando el aplicativo
- Fecha actual: despliega la fecha y hora actual del PC sobre el cual se está ejecutando el aplicativo
- Nombre del aplicativo: despliega el nombre del aplicativo que se está corriendo y la ruta sobre la cual se está ejecutando en la PC.



Figura 2.9

La barra de notificaciones fue diseñada con el objetivo de poder tener un registro de las máquinas sobre las cuales se están ejecutando los aplicativos y los usuarios que comúnmente ejecutan este aplicativo, además, esta barra de notificaciones va a permitir ayudar en el soporte a los usuarios.

Para el manejo de las tramas de comunicación entre el dispositivo Android y el lector NFC se tuvo que realizar una clase que permita activar el módulo NFC del dispositivo y opere de la manera que deseamos.

Esta programación fue realizada sobre Android empleando Eclipse ADT

A continuación se muestra el código que permite operar el módulo NFC de un teléfono Android con un sistema operativo 4.4

```
public class IsoDepTamaCommunicator extends AbstractTamaCommunicator {
    private Logger log = LoggerFactory.getLogger(getClass());

    private int messageCounter = 0;

    private static final byte[] CLA_INS_P1_P2 = { 0x00, (byte)0xA4, 0x04,
        0x00 };

    private static final byte[] AID_ANDROID = { (byte)0xF0, 0x01, 0x02, 0x
        03, 0x04, 0x05, 0x06 };

    public IsoDepTamaCommunicator(ByteStreamReader reader, ByteArray
        Writer writer) {
        super(reader, writer);
    }

    private byte[] createSelectAidAid(byte[] aid) {
        byte[] result = new byte[6 + aid.length];
        System.arraycopy(CLA_INS_P1_P2, 0, result, 0, CLA_INS_P1_P2.length);
        result[4] = (byte)aid.length;
        System.arraycopy(aid, 0, result, 5, aid.length);
    }
}
```

```

        result[result.length - 1] = 0;

        return result;
    }

    public void connectAsInitiator() throws IOException {
        while (true) {
            InListPassiveTargetResp inListPassiveTargetResp
=
                sendMessage(new InListPassiveTargetReq((
byte)1, (byte)0, new byte[0]));

            if (inListPassiveTargetResp.getNumberOfTargets(
) > 0) {

                log.info("TargetData: " +
                    NfcUtils.convertBinToASCII(inListPassiveTargetResp.getTargetData()));

                if (inListPassiveTargetResp.isIsoDepSupported()) {

                    log.info("IsoDep Supported");

                    byte[] selectAidApdu = createSelectAidApdu(AID_ANDROID);

                    DataExchangeResp resp =
                        sendMessage(
                            newDataExchangeReq(inListPassiveTargetResp.getTargetId(),
                                false, selectAidApdu, 0, selectAidApdu.length));

                    String dataIn = new String(resp.getDataOut());

                    log.info("Received: " + dataIn);

                    if (dataIn.startsWith("Hello")) {

                        exchangeData(inListPassiveTargetResp);
                    }
                }
            }
        }
    }

```

```

        }
    }

    else {
        log.info("IsoDep NOT Supported");
    }
    break;
}
else {
    try {
        Thread.sleep(100);
    }
    catch (InterruptedException e) {
        throw new RuntimeException(e);
    }
}
}

}

private void exchangeData(InListPassiveTargetRespinListPassive
TargetResp) throws IOException {
    DataExchangeRespresp;
    String dataIn;
    while (true) {
        byte[] dataOut = ("Message from desktop: " + me
ssageCounter++).getBytes();
        resp = sendMessage(

```

```

        newDataExchangeReq(inListPassiveTargetRe
sp.getTargetId(), false, dataOut, 0,
        dataOut.length));
        dataIn = new String(resp.getDataOut());
        log.info("Received: " + dataIn);
    }
}
}

```

El sistema desarrollado se integra con una base de datos desarrollada en PostgreSQL. A continuación se detalla el código de la clase que permite establecer la conexión a la base de datos a través de un JDBC.

```

public class ConnPostgres {
private static String url;
private static String user;
private static String password;
ConnPostgres(String url, String user, String password){
    this.url = url;
this.user = user;
this.password = password;
}
public static Connection conexion()
{
    Connection con = null;
    Statement st = null;
ResultSetrs = null;

```

```
ResultSet update = null;

try {

con = DriverManager.getConnection(url, user, password);

    } catch (SQLException ex) {

System.out.println("Error: " + ex);

    }

}

return con;

}
```

2.8.4. INSTALACIÓN Y CONFIGURACIÓN DE SOFTWARE BASE

Para el almacenamiento de datos se optó por incorporar una base de datos libre que tenga un buen desempeño reconocido por su tiempo en el mercado y confianza en los usuarios. La base de datos seleccionada es PostgreSQL, base de datos que nos ayudará también en el respaldo de los datos que maneje el aplicativo.

Es obligatorio para todo administrador de Base de Datos, sea SQL Server, Oracle, Access, MySQL, PostgreSQL, etc., realizar respaldo de la base de datos, más aun, crear una tarea programada que automáticamente realice esta tarea en un día y hora determinada.

Pero realmente lo correcto para realizar un respaldo de una BD es programando una tarea, que realice el respaldo automáticamente pautado en un día a una hora.

No es ninguna novedad el valor que tiene la información y los datos para los sistemas que ayudan a la automatización de controles, datos que pueden ayudar en la toma de decisiones de sus altos mandos o ayudar en análisis de la situación real de una organización. Es principalmente por esta razón, que se debe tener un esquema de respaldos programado para la información importante manejada en el sistema como es el caso de los accesos del personal.

De darse pérdidas de información se puede afectar en gran parte la toma de decisiones ante la aplicación de políticas de control

La tecnología no está exenta de fallas o errores, y los respaldos de información son utilizados como un plan de contingencia en caso de que una falla o error se presente.

Los Jobs son una herramienta extremadamente útil y que es utilizada para implementar cualquier tarea repetitiva de manera automática. Los Jobs pueden utilizarse para automatizar procesos en la base de datos, desde realizar respaldos de una base de datos o ejecutar un script en el servidor. Los Jobs básicamente ejecutan tareas programadas por el usuario en un cronograma definido.

Una vez finalizada la instalación el asistente nos dará la posibilidad de ejecutar StackBuilder, aplicación que nos permitirá instalar otros componentes y herramientas para PostgreSQL. En la figura 2.10 podemos observar la pantalla inicial que permite arrancar con la instalación de StackBuilder.



Figura 2.10: Inicio de instalación de StackBuilder

Si se realiza el marcado de la opción de instalación de StackBuilder, no permitirá seleccionar la instalación de la base de datos sobre la cual vamos a trabajar. Para el caso del proyecto seleccionamos "PostgreSQL 9.0 onport 5432" tal y como se muestra en la figura 2.11. Una vez seleccionada la opción correcta se procede a pulsar "Next".



Figura 2.11: Selección de Base de datos

Seleccionaremos las aplicaciones, componentes y herramientas a instalar y pulsaremos "Next". El componente que se desea instalar es el PGAGENT, por lo cual lo es necesario seleccionar esta opción tal y como se muestra en la figura 2.12. Una vez seleccionada la opción correcta se procede a pulsar "Next".

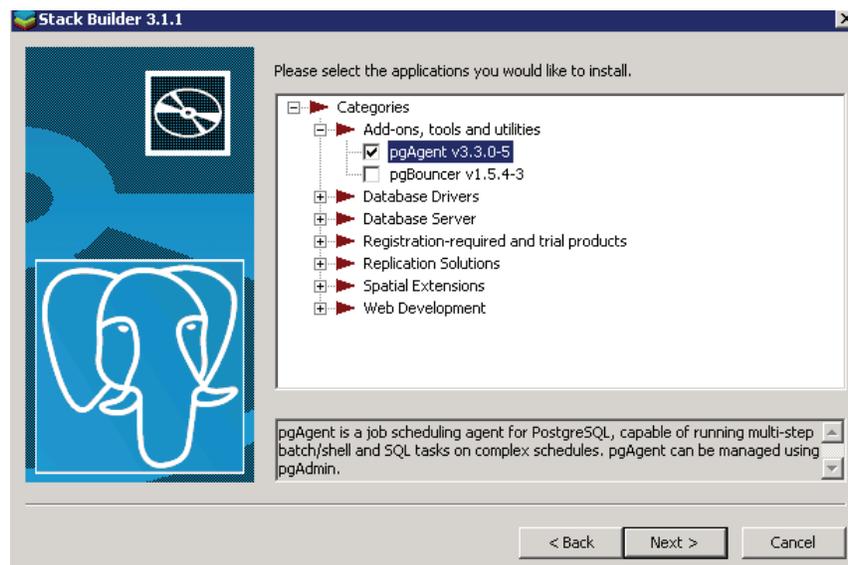


Figura 2.12: Selección de componentes adicionales

En la figura 2.13 se procede a seleccionar el directorio en el cual se va a realizar la descarga del componente específico. Por recomendación la ruta debe ser la establecida por defecto, una vez seleccionada la opción correcta se procede a pulsar "Next".

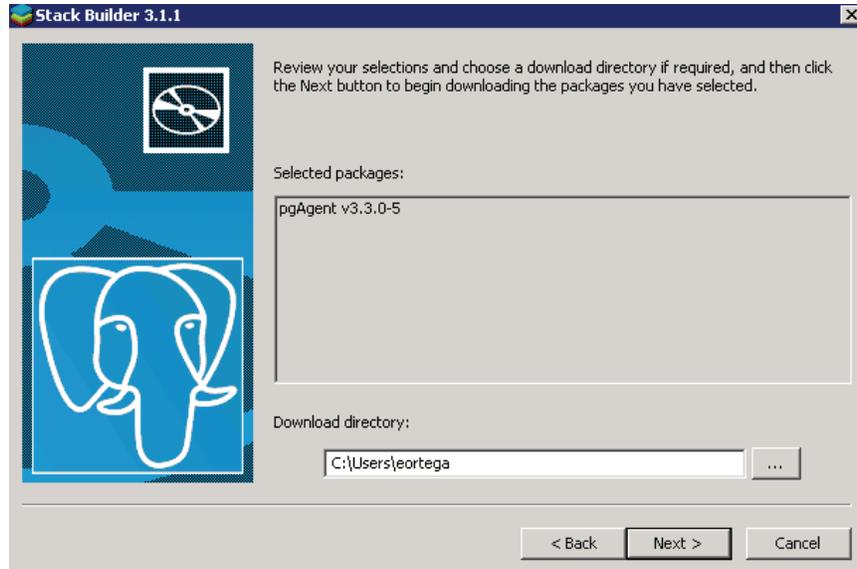


Figura 2.13: selección de directorio de instalación

En la figura 2.14 se procede a autorizar la instalación del componente. Una vez completada la descarga pulsamos “Next”

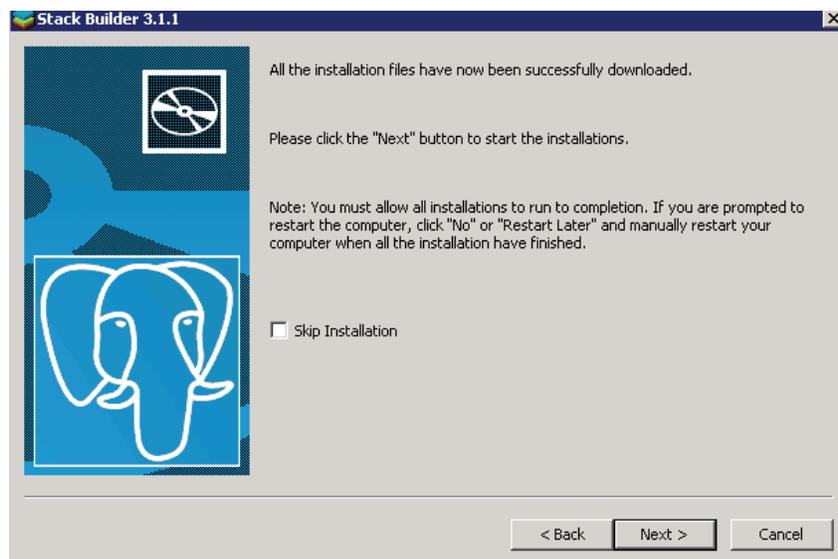


Figura 2.14: Autorización de instalación de componente

En la figura 2.15 el configurador de la instalación del componente se desplegará, pulsamos “Next”



Figura 2.15: Instalación de componente

En la figura 2.16 seleccionamos el directorio donde se va a proceder guardar los archivos de ejecución del componente.

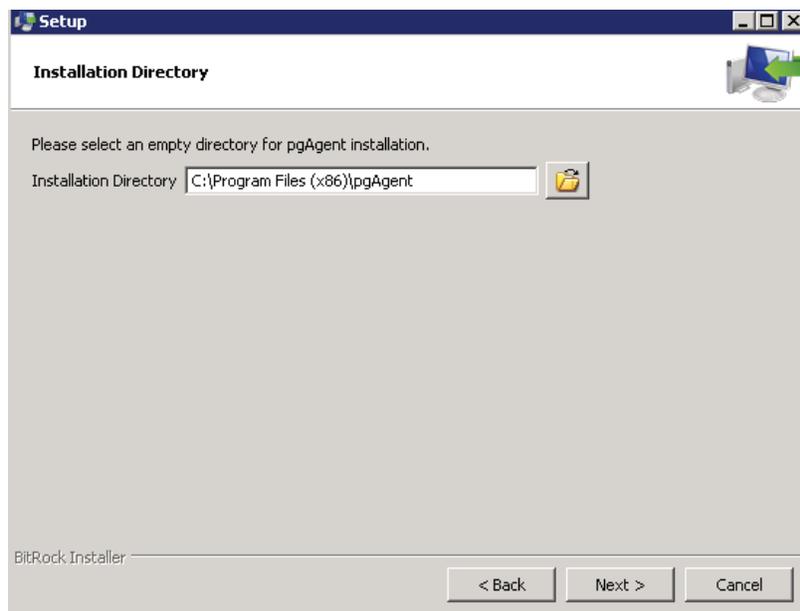


Figura 2.16: Selección de directorio para la instalación

En la figura 2.17 seleccionamos el modo de actualización que deseamos para administrador de la base de datos y sus componentes. Si se desea que las actualizaciones no alteren drásticamente la operación de la base de datos y se lleven de manera automática, seleccionamos "Upgrademode". Si la base es nueva es recomendable no seleccionar esta opción. Para nuestro caso no seleccionaremos el modo "UpgradeMode".

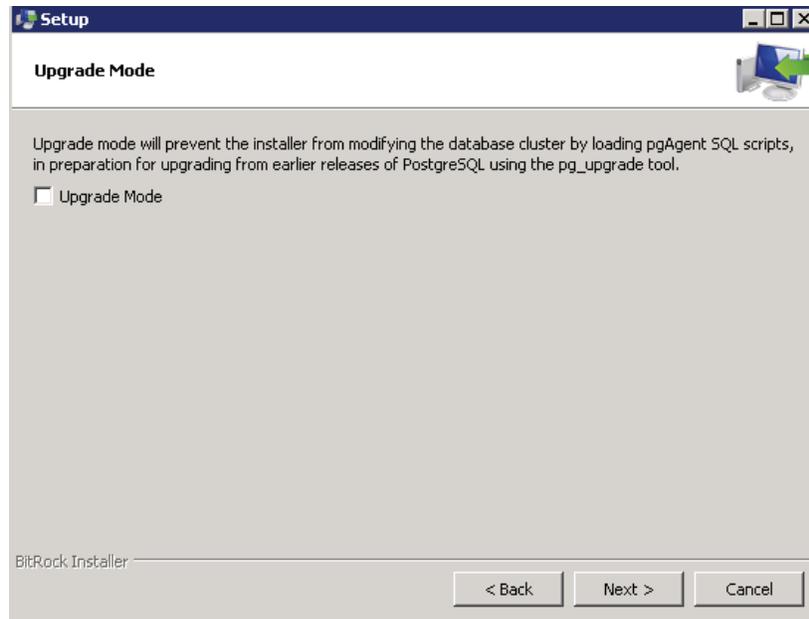


Figura 2.17: Actualizaciones automáticas

En la figura 2.18 ingresamos la clave del usuario principal de la base de datos, la dirección IP del host de la base de datos, el puerto de conexión y la contraseña del usuario especificado. Este usuario no necesariamente debe ser el usuario ROOT.

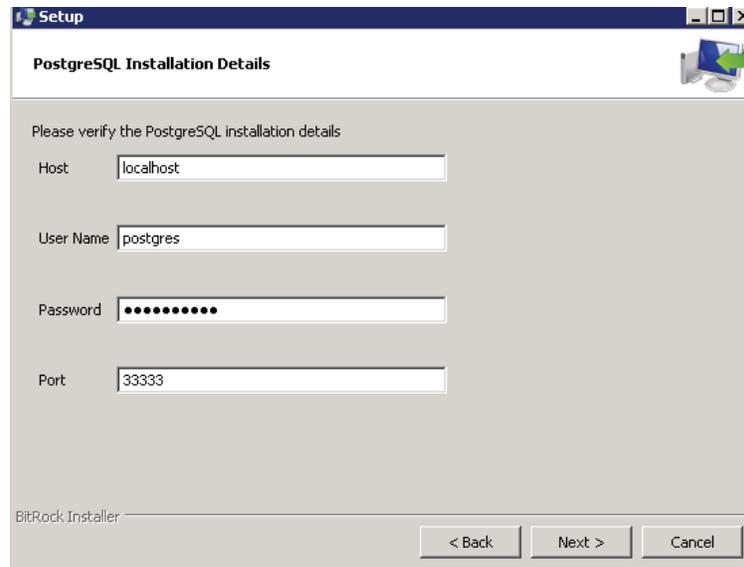


Figura 2.18: Datos de acceso a la base de datos

En la figura 2.19 visualizamos la ventana que nos permite crear la cuenta del usuario PgAgent. Esta cuenta administrará el arranque automático de los procesos, permitirá la creación de nuevos procesos, eliminación y actualización de procesos existentes.



Figura 2.19: Creación de usuario PgAgent

Una vez pulsado el botón “Next” de la figura 2.22 se procederá a instalar el componente con las configuraciones establecidas.

Es necesario realizar configuraciones básicas para habilitar la comunicación de la base de datos con PCs de manera remota, para esto es debemos ir al directorio de instalación de POSTGRESQL en el servidor, para nuestro caso nos ubicamos en la carpeta:

“C:\Program Files\PostgreSQL\9.3\data”

En esta carpeta es necesario enfocarse en dos archivos importantes:

- PG_HBA.CONF
- POSTGRESQL.CONF

El fichero PG_HBA.CONF se utiliza para definir como, donde y desde que sitio un usuario puede utilizar nuestro clusterPostgreSQL. Todas las líneas que empiecen con el caracter # son interpretadas como comentarios. El resto debe tener el siguiente formato:

[conexión] [base de datos] [usuario] [IP] [Mascara de red] [autenticación] [op]

Dependiendo del tipo de conexión y del tipo de autenticación, [IP],[mascara de red] y [opciones] pueden ser opcionales. El tipo de conexión puede tener los siguientes valores:

- local
- host
- hostssl

- hostnsssl

El tipo de método puede tener los siguientes valores:

- trust
- reject
- md5
- crypt
- password
- krb5
- ident
- pam
- ldap

En el archivo PG_HBA.CONF del servidor es necesario agregar la siguiente línea:

```
host all all 0.0.0.0/0 md5
```

En la figura 2.20 se puede observar la sección del archivo pg_hba que fue modificada para poder permitir conexiones remotas desde cualquier IP.

```
# TYPE DATABASE USER ADDRESS METHOD
# IPv4 local connections:
host all all 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
#host replication postgres 127.0.0.1/32 md5
#host replication postgres ::1/128 md5
host all all 0.0.0.0/0 md5
```

Figura 2.20: Archivo pg_hba.conf

Los cambios que realicemos en el fichero PostgreSQL.CONF afectarán a todas las bases de datos que tengamos definidas en nuestro cluster PostgreSQL. La mayoría de los cambios se pueden poner en producción con un simple 'reload', otros cambios necesitan que arranquemos de nuevo nuestro cluster.

En el archivo PostgreSQL.CONF se almacenan parámetros de comunicación de la base de datos, entre los más importantes y relevantes para el caso son los siguientes:

- listen_addresses
- port

LISTEN_ADDRESSES indica la dirección IP que se va a conectar de manera remota con Postgres, para el caso ese parámetro tiene el valor (*), esto indica que cualquier IP puede conectarse a postgres

PORT indica el puerto de comunicación con la base de datos, para el caso tenemos este parámetro con el valor (33333).

Hay que recordar que cualquier cambio que se haga en alguno de estos dos archivos para que tome efecto, es necesario reiniciar el servicio de postgres desde los servicios de Microsoft como se muestra en la figura 2.24.

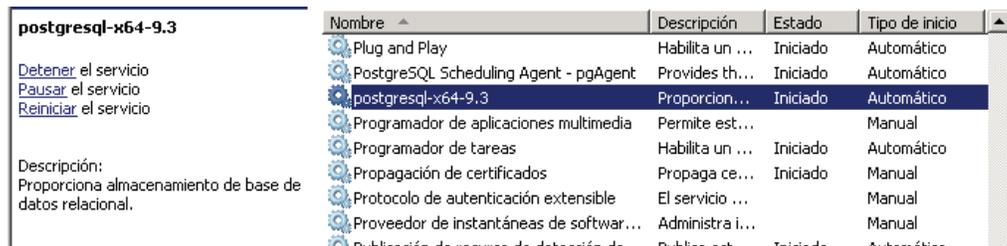


Figura 2.21: administración de servicios de Windows

Es recomendable antes de realizar la instalación del sistema de firmas electrónicas realizar una prueba de conexión de la base de datos desde la máquina donde se va a instalar el firmador electrónico.

2.8.4.1. Creación de tareas programadas

Esto permitirá borrar registros lógicamente de ciertas tablas que consideremos prudentes por cuestiones de mantenimiento.

Citando como ejemplo para comprender de mejor manera, un sistema de facturación electrónica, cualquiera, tiene tres tablas cuyo tamaño a largo plazo puede convertirse en un factor crítico, ya que en estas tablas se almacenan texto en formato XML cuyo tamaño va desde los 12 KB hasta los 15 KB. Las tablas son las siguientes:

- XML_FIRMAR: almacena los documentos XML pendientes de firmar.
- XML_FIRMADOS: almacena los documentos XML firmados pendientes de envío.
- RESPUESTAS_SRI: almacena los XML que entrega como respuesta el SRI.
- XML_CONTINGENTES: almacena los XML emitidos en contingencia.

Es necesario crear tareas programadas de mantenimiento sobre cada una de las tablas antes mencionadas.

Sobre nuestro sistema de control de acceso, es necesario que el personal del área de sistemas analicé sobre que tablas puede ser prudente correr estas tareas programadas para evitar posibles colapsos a futuro.

Para que estas tareas sean ejecutadas sin inconvenientes es necesario realizar configuraciones iniciales. Es necesario ubicar el archivo PGPASS.CONF que

siempre es instalado en la carpeta de usuarios del sistema (“C:\Users\postgres\AppData\Roaming\postgresql”).

Al abrir el archivo es muy probable que se encuentre vacío o puede darse que este parcialmente lleno. El archivo debe de contar con las siguientes líneas (mínimo):

- localhost:33333:postgres:postgres:TECNO.2013
- localhost:33333:createc:postgres:TECNO.2013

Cada uno de los parámetros se especifican en la tabla 2.25.

Tabla 2.25: Parámetros de conexión

Parámetros	Descripción
LOCALHOST	IP o host de la base de datos donde se va a ejecutar la tarea programada
33333	Puerto de comunicación
POSTGRES	Base de datos sobre la cual se ejecutarán las tareas programadas
POSTGRES	Usuario con acceso a la base de datos (permisos de DBA)
TECNO.2013	Clave del usuario

Comprendido los parámetros involucrados se procede a agregar esas líneas en el archivo, una de ellas conectando a la base de datos principal (postgres) y la otra conectando a la base de datos sobre la cual se van a ejecutar las tareas programadas. Al final tendremos un archivo muy similar a lo que se muestra en la figura 2.22.

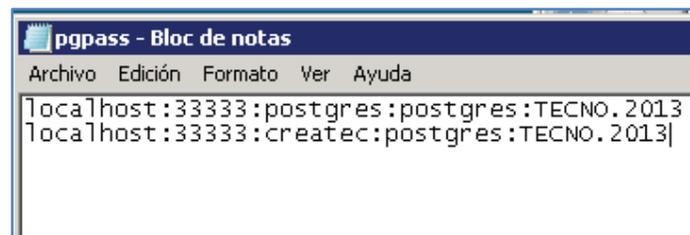


Figura 2.22: fichero pgpass

Una vez almacenados los cambios efectuados en el archivo se procede a reiniciar el servicio de manejo de tareas programadas en POSTGRESQL llamado “PostgreSQLSchedulingAgent - pgAgent”

Dentro de las instalaciones de programas base, es necesario la instalación de un driver que no es nada más que un programa informático que permite al sistema operativo interactuar con el periférico, haciendo una abstracción del hardware y proporcionando una interfaz para usarlo.

Se puede esquematizar como un manual de instrucciones que indica cómo debe controlar y comunicarse con un dispositivo en particular. Por tanto, es una pieza esencial, sin la cual no se podría usar el lector NFC.

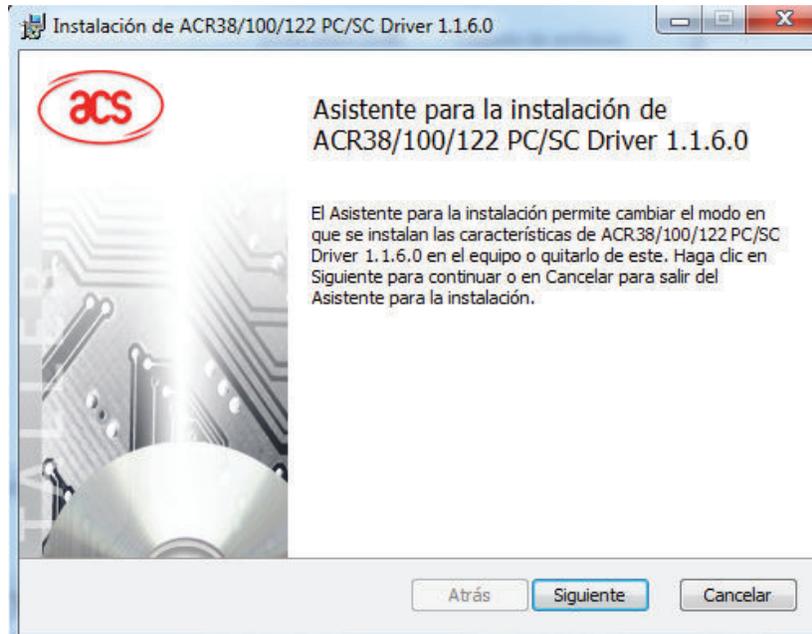


Figura 2.23: Driver ACR

CAPÍTULO 3. IMPLEMENTACIÓN, PRUEBAS Y ANÁLISIS DE RESULTADOS

3.1. IMPLEMENTACIÓN

Para la implementación del sistema de control de accesos en la organización, es necesario preparar, lo que será, nuestro entorno de producción.

El objetivo de la preparación del entorno de producción es planificar que todos los recursos estén disponibles para la puesta en producción de los Sistemas de Información.

En esta tarea, es necesario disponer de todos los recursos necesarios para realizar la puesta en producción de los componentes y subsistemas que conforman el sistema de información. Para ello, se asegura la disponibilidad del entorno y de los datos necesarios para ejecutar la puesta en producción, se preparan las bibliotecas o librerías que se estimen oportunas para la realización de las mismas, así como los procedimientos manuales o automáticos asociados, conforme a la especificación del paso a producción.

En la tabla 3.1 se nombra la tarea que se llevó a cabo junto con una breve descripción.

Tabla 3.1: Tareas programadas

	Tarea	Descripción
1	Instalación de base de datos PostgreSQL	Instalación de PostgreSQL 9.3 Instalación de PgAgent Instalación de complementos de base de datos
2	Configuración de base de datos	Configuración de parámetros de conexión remota Configuración de acceso remoto Configuración de PgAgent (tareas programadas)
3	Levantamiento de base de datos	Creación de base de datos de prueba Creación de tareas programadas Ejecución de Scripts
4	Prueba de conexión	Prueba de conexión desde el PC donde se instalará el aplicativo de firma electrónica, hasta el servidor de base de datos PostgreSQL
5	Instalación de JAVA	Pruebas ejecución de tareas programadas Instalación de JDK 8

Instalación de JAVA FX 2.2.7		
9	Instalación de aplicativo	Instalación del aplicativo de control de accesos
10	Configuración del aplicativo	Creación del archivo de configuración
11	Registro de lectores en el sistema	Empleando la interfaz del sistema se procederá a la creación de registros
12	Registro de empresa en el sistema	Empleando la interfaz del sistema se procederá a la creación de registros
13	Registro de áreas en el sistema	Empleando la interfaz del sistema se procederá a la creación de registros

3.1.1. INSTALACIÓN DEL APLICATIVO DE CONTROL DE ACCESOS

Reuniones continuas con el personal del área de sistemas de la organización, permitió llegar a acuerdos sobre los recursos que emplearemos para la instalación del aplicativo.

En la tabla 3.2 se detallan los directorios a ser empleados para la instalación del aplicativo. Estos directorios almacenarán información vital para el correcto funcionamiento del aplicativo.

Tabla 3.2: Instalación del aplicativo

Equipo de instalación	CTF10
IP	10.10.40.10
Directorio aplicativo	C:\Program Files (x86)\AccessMIT
Almacén de certificados	C:\Program Files\Java\jre8\lib\security\cacerts
Directorio JAVA	C:\Program Files\Java\jre8\bin
Directorio log de errores	C:\Program Files (x86)\AccessMIT\log
Directorio JAVAW	C:\Program Files\Java\jre8\bin\javaw.exe

3.1.2. BASE DE DATOS POSTGRESQL 9.3

Reuniones continuas con el personal del área de sistemas de la organización, permitió llegar a acuerdos sobre los recursos que se emplearán para la instalación de la base de datos seleccionada para el proyecto.

En la tabla 3.3 se detalla el nombre del servidor en el cual se instalará la base de datos junto con la IP, además se detalla el puerto que será empleado y las credenciales del usuario ROOT.

Tabla 3.3: Instalación de BDD

Equipo de instalación	SERVER.MIT.COM
IP	10.10.40.20
Puerto	5432
USR	Postgres
PSW	DE_MIT
Esquema de datos (base)	mit_pruebas

3.1.3. ARCHIVO DE CONFIGURACIÓN

El archivo de configuración permite cargar parámetros importantes para el funcionamiento del aplicativo de control de accesos. Este archivo cuenta con una estructura específica de la cual se leen parámetros para ser usados por el aplicativo de control de accesos. Dentro de este archivo se especificaran parámetros que pueden ser comúnmente modificados por los administradores del aplicativo. A continuación se detallará la estructura y la descripción de cada uno de los campos que forman parte del archivo de configuración

El archivo config.ini deberá estar ubicado en el mismo directorio en el cual se carga el ejecutable del aplicativo (controlACC.jar), caso contrario el aplicativo no se ejecutará.

El archivo de configuración consta de 3 secciones:

- [MIT]: contiene parámetros de conexión con la base de datos, certificados , autorización de uso del aplicativo y directorios necesarios de lectura y escritura.
- [EMPRESAS]: guarda el licenciamiento de cada una de las empresas autorizadas para controlar los accesos de su personal. Cada empresa registrada en esta sección podrá registrar usuarios.

3.1.3.1. Sección [MIT]

Esta sección cuenta con los siguientes parámetros:

- Autorización: nombre de la empresa a la cual se le autoriza el uso del aplicativo

- Producto: sentencia empleada para la validación de la clave del producto, esta sentencia a su vez se compone de tres partes que son:
 - Código de empresa
 - RUC de la empresa
 - Fecha de implementación
- Clave: clave de validación del producto
- PATH-LOG: directorio que requiere permisos de escritura en el cual se escribirá un log que ayudara a comprender el funcionamiento del aplicativo. Se creará un archivo .txt por cada inicio de sesión del aplicativo.
- PATH-NA: directorio que requiere permisos de escritura en el cual se almacenarán los registros de los accesos que hayan fallado en la autorización.
- URL: parámetro de conexión de la base de datos, ej:jdbc:postgresql://10.10.40.20:5432/mit_pruebas, este parámetro se puede obtener desde el aplicativo de prueba de conexión con la base de datos.
- USR: usuario con control total a la base de datos específica.
- PSW: clave del usuario.

3.1.3.2. Sección [empresas]

Esta sección cuenta con los siguientes parámetros:

- Autorización: nombre de la empresa a la cual se le autoriza el uso del aplicativo
- Producto: sentencia empleada para la validación de la clave del producto, esta sentencia a su vez se compone de tres partes que son:
 - Código de empresa
 - RUC de la empresa
 - Fecha de implementación
- Clave: clave de validación del producto

Esta sección debe de tener por lo menos una empresa. Es en esta sección donde se cargan las empresas que el sistema va a controlar. Si una empresa no se encuentra en esta sección, el aplicativo no procederá a registrar el acceso de usuarios pertenecientes a dicha empresa.

A continuación se ilustra la estructura de un archivo de configuración empleado para la empresa MIVSELLTECH.

```
[MIT]
AUTORIZACION = MIVSELL S.A.
PRODUCTO = SD/1791434374001/2014-AGO-01
CLAVE = iXAQD/m5/j3169IJGD
PATH-LOG = D:\\log
PATH-NA = D:\\NO_ACC
```

```
URL = jdbc:postgresql://10.10.40.10:33333/MIT_PRUEBAS
USR = postgres
PSW = DE_MIT
```

```
[EMPRESAS]
AUTORIZACION = MIVSELL S.A.
PRODUCTO = SD/1791434374001/2014-AGO-01
CLAVE = iXAQD/m5/j3169IJGD
```

3.1.4. AUTORIZACIÓN

La reproducción de copias de software sin las correspondientes autorizaciones, es un problema que se tiene a diario en nuestro entorno. Es por esta razón que se ha buscado un método básico de control de la autorización de uso del producto de software que se desarrolló.

La autorización del producto tiene con finalidad de establecer un control inicial que evite el uso indebido del software por personal externo a la organización.

La autorización del producto de software se compone de 3 partes:

- Autorización
- Producto
- Clave

El campo autorización debe contener la razón social de la organización autorizada para el uso del aplicativo. El producto es una secuencia de caracteres que se compone del RUC (registro único de contribuyente) de la organización autorizada, el módulo al cual corresponda el aplicativo y la fecha en la cual se autoriza el uso del aplicativo a la organización. Esta estructura se la puede observar en la tabla 3.4.

Tabla 3.4: Autorización del producto

AUTORIZACION	MIVSELL
PRODUCTO	1391786549001/ACC/2014-MAY-02
CLAVE	t8eiO6+Xi1/bkJA11/X5QK31

3.2. PRUEBAS

3.2.1. PRUEBAS FUNCIONALES

A este tipo de pruebas se les denomina también pruebas de comportamiento o pruebas de caja negra, ya que los testers o analistas de pruebas, no enfocan su atención a como se generan las respuestas del sistema, básicamente el enfoque de este tipo de prueba se basa en el análisis de los datos de entrada y en los de salida, esto generalmente se define en los casos de prueba preparados antes del inicio de las pruebas.

Estas pruebas estuvieron a cargo del personal operativo del sistema, esto implica la recepcionista de la empresa y personal del área técnica.

Estas pruebas fueron llevadas a cabo usando matrices de funcionalidad, las cuales tienen como objetivo llegar a los casos de éxito y casos alternativos más críticos de los módulos del sistema.

Cada matriz busca llegar a un objetivo, el cual lo alcanzamos a través de:

- A. Requisito (Condición): es una norma que se debe de cumplir para que las pruebas sean efectivas.
- B. Pasos: serie de tareas que se deben de seguir para alcanzar el objetivo
- C. Datos de prueba: son datos que se emplearan en cada uno de los pasos, estos datos deben ser lo más reales posibles para que las pruebas sean efectivas.
- D. Resultado: es la respuesta obtenida después de haber seguido cada uno de los pasos determinados

Estas pruebas ayudaron en gran parte a comprender el comportamiento del aplicativo ante situaciones reales. Los usuarios son parte esencial de este proceso ya que es aquí donde se lograron aclarar dudas y mal entendidos que pueden haber con respecto a los procesos a seguir para operar el sistema.

Durante este proceso de pruebas se procuró tener la completa atención de los involucrados, esto implicaba, buscar el día y el momento adecuado en el que se pueda contar con la completa atención de los usuarios e involucrados en la prueba.

Para este proceso de pruebas se tuvo la participación de cuatro personas; recepcionista de la empresa, gerente de proyectos, asistente técnico y desarrollador del aplicativo.

Las pruebas funcionales se llevaron a cabo sobre los siguientes módulos:

- Módulo de empresas
- Módulo de empleados
- Módulo de áreas
- Módulo de lector

Sobre cada una de estas pruebas se buscó llegar a los objetivos establecidos, que son principalmente, realizar el registro de empresas, empleados, áreas y lectores en el sistema.

En la tabla 3.5 podemos observar las pruebas efectuadas en el módulo de empresas.

En la tabla 3.6 podemos observar las pruebas efectuadas en el módulo de empleados.

En la tabla 3.7 podemos observar las pruebas efectuadas en el módulo de áreas.

En la tabla 3.8 podemos observar las pruebas efectuadas en el módulo de lectores.

El objetivo principal de llevar a cabo estas pruebas es ver la reacción de los usuarios finales ante la interacción con el sistema. Estas pruebas nos servirán, de igual manera, como estrategias de recopilación de información relevante como facilidad de uso, diseño del aplicativo, tiempo de respuesta, entre otros.

Fue primordial estar atentos a las reacciones que tienen los usuarios, ya que estas reacciones nos dieron pistas básicas sobre la aceptación de los usuarios al sistema, o si hubiese sido el caso, estas reacciones nos hubieran dado pistas sobre la resignación o disgusto de los usuarios al uso del sistema. (Balaguera, 2013)

3.2.1.1. Pruebas Módulo de empresas

Tabla 3.5: prueba de módulo de empresas	
Nombre del proyecto	NFC - MóduloEmpresas
Fecha de creación	04-ago-14
Fecha de revisión	29-oct-14

ID Prueba	Objetivo Principal	Condición	Pasos:	Datos de prueba	Resultado esperado	Resultado Obtenido
TF_ME_01	Registro exitoso de una empresa	1. El registro debe de ser realizado por la recepcionista del edificio	<ol style="list-style-type: none"> Ingrese un nombre de empresa Ingrese el RUC de la empresa Ingrese horas de salida y de llegada 	<p>"MIVSELL"</p> <p>"1714328638001"</p> <p>9:00:00 18:00</p>	La empresa ha sido registrada de manera exitosa	La empresa ha sido registrada de manera exitosa
TF_ME_02	Mensaje de error al registrar una empresa con mismo número de RUC	<ol style="list-style-type: none"> El registro debe de ser realizado por la recepcionista del edificio Se debe tener una empresa registrada 	<ol style="list-style-type: none"> Ingrese un nombre de empresa Ingrese el RUC de la empresa Ingrese horas de salida y de llegada 	<p>"CREATEC"</p> <p>"1714328638001"</p> <p>9:00:00 18:00</p>	Error en el registro de la empresa. Empresa ya registrada	Error en el registro de la empresa. Empresa ya registrada

3.2.1.2. Pruebas Módulo Empleados

Tabla 3.6: prueba de módulo de empleados	
Nombre del proyecto	NFC - Módulo Empleados
Fecha de creación	04-ago-14
Fecha de revisión	29-oct-14

ID Prueba	Objetivo Principal	Condición	Pasos:	Datos de prueba	Resultado esperado	Resultado Obtenido
TF_MEM_01	Registro exitoso de una empleado	1. El registro debe de ser realizado por la recepcionista del edificio	<ol style="list-style-type: none"> Ingrese un nombre de empleado Ingrese el apellido del empleado Ingrese el ID del empleado ingrese el PIN del empleado Ingrese la empresa a la cual pertenece el empleado 	<p>"ERIKA"</p> <p>"MOLINA"</p> <p>"1714328638"</p> <p>"834E45C900001"</p> <p>"MIVSELL"</p>	El empleado ha sido registrada de manera exitosa	El empleado ha sido registrada de manera exitosa

TF_MEM_02	Mensaje de error al registrar un empleado con mismo id	<p>1. El registro debe de ser realizado por la recepcionista del edificio</p> <p>2. Se debe tener un empleado registrado</p>	6. Ingrese las horas de ingreso y salida del empleado	9:00:00 18:00		
			1. Ingrese un nombre de empleado	"JUAN"	Error en el registro de empleado. Empleado ya registrado	Error en el registro de empleado. Empleado ya registrado
			2. Ingrese el apellido del empleado	"TTORRES"		
			3. Ingrese el ID del empleado	"1714328638"		
			4. Ingrese el PIN del empleado	"834E45C900001"		
			5. Ingrese la empresa a la cual pertenece el empleado	"MIVSELL"		
			6. Ingrese las horas de ingreso y salida del empleado	9:00:00 18:00		

3.2.1.3. Pruebas Módulo áreas

Tabla 3.7: prueba de módulo de áreas	
Nombre del proyecto	NFC - Módulo Áreas
Fecha de creación	04-ago-14
Fecha de revisión	29-oct-14

ID Prueba	Objetivo Principal	Condición	Pasos:	Datos de prueba	Resultado esperado	Resultado Obtenido
TF_MEM_01	Registro exitoso de una área	1. El registro debe de ser realizado por la recepcionista del edificio	<ol style="list-style-type: none"> Ingrese un nombre de área Ingrese el código de área Ingrese la descripción de área Ingrese el lector asignado 	<p>"RECEPCION"</p> <p>"RCP"</p> <p>"ENTRADA PRINCIPAL"</p> <p>"ACR122"</p>	El área ha sido registrada de manera exitosa	El área ha sido registrada de manera exitosa
TF_MEM_02	Mensaje de error al registrar un área con mismo id	<ol style="list-style-type: none"> El registro debe de ser realizado por la recepcionista del edificio Se debe tener un área registrada 	<ol style="list-style-type: none"> Ingrese un nombre de área Ingrese el código de área 	<p>"RECEPCION2"</p> <p>"RCP"</p>	<p>Error en el registro del área. Área ya registrada</p> <p>Error en el registro del área. Área ya registrada</p>	<p>Error en el registro del área. Área ya registrada</p> <p>Error en el registro del área. Área ya registrada</p>

		3. Ingrese la descripción de área	"ENTRADA PRINCIPAL"	
		4. Ingrese el lector asignado	"ACR122"	

3.2.1.4. Pruebas Módulo lector

Tabla 3.8: prueba de módulo de lector	
Nombre del proyecto	NFC - Módulo Lectores
Fecha de creación	04-ago-14
Fecha de revisión	29-oct-14

ID Prueba	Objetivo Principal	Condición	Pasos:	Datos de prueba	Resultado esperado	Resultado Obtenido
TF_ML_01	Registro exitoso de un lector NFC	1. El registro debe de ser realizado por la recepcionista del edificio	1. Ingrese un modelo de lector 2. Ingrese un ID del lector 3. Ingrese el fabricante del lector 4. Ingrese la descripción del lector	"ACR122" "ID_A" "ASUS" "LECTOR DE PRUEBA"	El lector ha sido registrado de manera exitosa	El lector ha sido registrado de manera exitosa

3.3. ANÁLISIS DE RESULTADOS

El tiempo total invertido en el proceso de pruebas se redujo considerablemente con el modelo RAD como los prototipos se prueban de forma independiente durante cada iteración. Sin embargo necesitan el flujo de datos y las interfaces entre todos los componentes para ser probado a fondo con cobertura de la prueba completa. Dado que la mayoría de los componentes de programación ya se han probado, reduce el riesgo de todas las cuestiones importantes.

Las pruebas de funcionalidad fueron llevadas a cabo conjuntamente con los usuarios finales y el personal técnico con el fin de comprender de mejor manera las necesidades, problemas o posibles inconvenientes que el usuario pueda tener con la interfaz, ingreso de datos o despliegue de resultados.

Dentro de la organización se evaluaron los siguientes aspectos:

- Facilidad de Instalación
- Facilidad de uso del sistema
- Diseño de interfaz
- Apoyo de la herramienta en el proceso

Para poder evaluar cada uno de estos aspectos se llevó a cabo una encuesta en la cual participó personal de la empresa. Dicho personal fue seleccionado según los siguientes criterios:

- El personal debe de tener como mínimo un año de experiencia laborando en la empresa.
- El personal debe de contar con un dispositivo móvil inteligente con sistema operativo Android y tecnología NFC
- El personal debe de tener disponibilidad de tiempo en la empresa con el fin de conocer a detalle las necesidades e inquietudes relacionadas con el sistema.

La encuesta será de carácter anónimo sin importar el género del encuestado, nivel de educación o tipo de profesión que ejerce dentro de la organización.

De acuerdo a los criterios antes mencionados, se seleccionó un grupo de 12 personas dentro del cual existe personal del área administrativa, área de sistemas informáticos y departamento de ventas.

El personal encuestado tuvo que responder una serie de preguntas cerradas, seleccionando una de las opciones dadas. Las preguntas a responder fueron:

1. Con respecto al proceso de instalación del aplicativo, considera que esta instalación fue:
 - a. Poco difícil
 - b. Difícil

- c. Muy difícil
2. Con respecto al uso de la aplicación, considera que el sistema es:
 - a. fácil de operar
 - b. difícil de operar
 3. Cree usted que el sistema ayuda a satisfacer los registros de ingreso y salida del personal a la empresa?
 - a. si
 - b. no
 4. Con respecto al diseño de la aplicación, considera que el aplicativo tiene una interfaz:
 - a. poco amigable
 - b. amigable
 - c. muy amigable
 5. Con respecto a la disponibilidad en el mercado de teléfonos inteligentes con NFC, considera que adquirir un teléfono inteligente con NFC es:
 - a. poco complicado
 - b. complicado
 - c. muy complicado
 6. Considera que el rendimiento del sistema es
 - a. bueno
 - b. regular
 - c. malo
 7. Comparando con otros sistemas de control de acceso que emplean tarjetas electromagnéticas, usted considera que reemplazar las tarjetas electromagnéticas por teléfonos inteligentes es:
 - a. beneficioso
 - b. no beneficioso
 - c. le da igual
 8. Considera que el sistema incorpora todo lo necesario para controlar los accesos del personal
 - a. si
 - b. no
 9. Considera usted que el sistema cubre las necesidades básicas de la mayoría de empresas
 - a. si
 - b. no
 10. Qué opinión en general tiene con respecto al sistema?
 - a. bueno
 - b. regular
 - c. malo

Respecto a la facilidad o dificultad de instalación del sistema NFC, el 90% de los usuarios encuestados considera que es fácil el proceso de instalación, luego de haber recibido capacitación. Apenas un 3% considera que es muy

difícil instalar el aplicativo, mientras que para el 7% de los encuestados encontraron complicaciones durante el proceso de instalación.



Respecto a la facilidad o dificultad de operar con el sistema, el 86% de los usuarios encuestados considera que es fácil operar con el mismo, luego de haber recibido capacitación. Apenas un 3% considera que es difícil utilizar el sistema de control de acceso, mientras que para el 11% lo manejaron de una manera intuitiva.



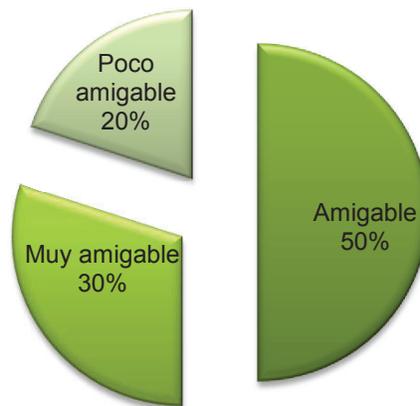
el 80% de los usuarios del sistema NFC, está (al menos) satisfecho con el sistema como herramienta de apoyo informático para establecer controles en el ingreso y salida de personal. En el mismo sentido, el 20% muestra un grado de

insatisfacción y consideran que el sistema no ayudará a controlar los accesos del personal de la empresa. La razón principal por la cual consideran que no es una herramienta que va a apoyar al control de acceso del personal es que depende de tecnología móvil. Consideran que la adquisición de los dispositivos móviles resulta costosa, además, consideran que el cambio de tecnología podría causarles molestias o decepciones.



5 de cada 10 usuarios del sistema NFC, está (al menos) satisfecho con el diseño de la interfaz de usuario considerándola amigable para el usuario, mientras que el 30% considera que la interfaz es muy amigable, sin embargo, que se podía incorporar mejores elementos de diseño, y por último, el 20% considera que la interfaz es poco amigable para el usuario.

Interfaz de usuario



El 50% de los encuestados, está consciente de que adquirir un teléfono inteligente con NFC es poco complicado, mientras que el 40% considera que adquirir un teléfono con NFC es complicado debido al poco conocimiento que se tiene con respecto a la tecnología NFC, y por último, únicamente el 10% de los encuestados considera que es muy complicado adquirir un teléfono con NFC debido a los costos de los equipos en el mercado. Dentro de este 10% de los encuestados, se encuentran personas que muestran un cierto grado de resistencia al cambio de tecnología.

Aquirir teléfonos con NFC



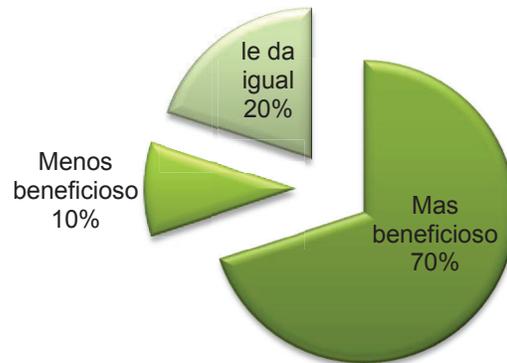
Con respecto al rendimiento del sistema, el 60% de los encuestados considera que el rendimiento del sistema es bueno en la máquina donde se instaló el

aplicativo, el 20% considera que el rendimiento del aplicativo es regular mientras que el otro 20% tuvo inconvenientes al desplegar otras aplicaciones por lo que considera que el rendimiento del aplicativo es malo al convivir con otras aplicaciones comunes en los equipos.



Con respecto al uso de teléfonos inteligentes en reemplazo de las tarjetas electromagnéticas, el 70% de los encuestados considera que bueno integrar la tecnología NFC en los teléfonos, el 10% considera que más comodidad le da el uso de las tarjetas electromagnéticas por lo cual lo ve menos beneficios y el otro 20% le da igual el uso de teléfonos con NFC y tarjetas electromagnéticas. Cabe mencionar que dentro de este 20% de los encuestados, existen personas que nunca han sido usuarios de sistemas que controlen el acceso del personal.

Beneficios de usar NFC en teléfonos inteligentes



Respecto a que si el sistema incorpora todo lo necesario para controlar los accesos del personal, el 70% de los encuestados considera que el sistema si cuenta con todo lo necesario para controlar los accesos del personal, mientras que el 30% considera que el sistema no cuenta con lo esperado por un sistema de control de accesos. Entre las funcionalidades que se deberían de incluir, lo encuestados sugirieron que se debe entregar alertas de los accesos de una manera más dinámica y efectiva como alerta a través de correos electrónicos o mensajes de textos.

El sistema incorpora todo lo necesario para controlar el acceso del personal?



Con respecto a que si el sistema cubre las necesidades básicas de la mayoría de las empresas, el 76% de los encuestados supo manifestar que si cubriría las

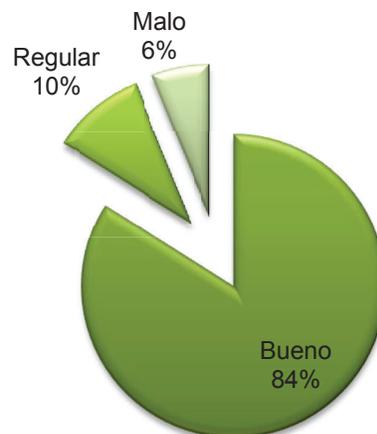
necesidades básicas. El 24% restante manifestó que el sistema no cubre las necesidades básicas de las empresas ya que ven como necesario e indispensable que el sistema pueda integrarse con aplicativos de administración más complejos, como ERPs.

El sistema cubre las necesidades básicas de las empresas?



El 84% de los encuestados opina que el sistema es bueno y es útil para la empresa, el 10% tiene una opinión regular sobre el sistema ya que consideran que se pudo haber dado más realce a la interacción del usuario con el dispositivo móvil, y el 6% tiene un concepto malo sobre el sistema ya que consideran que el uso de dispositivos móviles de precio elevado ocasionaría que el sistema sea poco empleado o comercializado.

Opinión general sobre el sistema de control de acceso



Como resultado de la encuesta realizada en la organización se concluye que el sistema satisface las necesidades de los usuarios y que cuenta con una gran aceptación por parte del personal de la organización, por estas razones, el sistema es considerado apto para ser usado como herramienta de control de acceso del personal a la organización.

Se ha impartido el conocimiento necesario a los usuarios y operadores del sistema con el fin de que este pueda reaccionar ante cualquier imprevisto o novedad que se presente y poder administrar de manera adecuada el sistema.

CAPÍTULO 4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- Con la metodología RAD se pudieron alcanzar las metas propuestas en el proyecto, reduciendo costos y tiempo sin afectar a la calidad final del producto.
- El uso de la tecnología NFC permitió obtener un sistema mucho más eficiente, ya que se basa en la lectura de tarjetas por proximidad, sin que el usuario tenga que acercarse a la lectora, como sucede con la tecnología RFID.
- Mediante el empleo de herramientas CASE se logró acelerar el diseño tanto de la base de datos así como también de la interfaz del sistema.
- Con la integración de la tecnología NFC y el sistema operativo Android se pudo potenciar el uso de los teléfonos inteligentes sin afectar el desempeño de la batería del equipo o afectando el desempeño de las aplicaciones comúnmente usadas.
- La combinación de la tecnología NFC con Android ayuda a mejorar la seguridad, ya que minimiza el riesgo de clonación de los datos encriptados, como sucede con las tarjetas magnéticas empleadas con frecuencia en los sistemas de control de acceso convencionales.

4.2. RECOMENDACIONES

- Se recomienda que el sistema de control de acceso sea operado por personal capacitado que pertenezca a la organización, con el fin de que este pueda reaccionar ante cualquier eventualidad que se presente durante el funcionamiento del aplicativo.
- Se recomienda que se definan políticas de mantenimiento del aplicativo que deben de ser elaboradas por personal con conocimientos de informática.
- Es recomendable monitorear por un tiempo prudente el funcionamiento del aplicativo y documentar cualquier novedad que se tenga con respecto a la operación o uso del sistema.
- Se recomienda adquirir un servidor exclusivo y dedicado a alojar la base de datos del aplicativo de control de acceso, ya que la información que este aplicativo procese puede, a futuro, causar inconvenientes de desempeño en las demás aplicaciones alojadas en el servidor.
- Se recomienda seguir la metodología RAD para realizar cualquier cambio al aplicativo, evitando incomodidades durante la ejecución de los cambios.

BIBLIOGRAFÍA

- [1] Advanced Card Systems. (s.f.). *Advance Card Systems*. Recuperado el 1 de Mayo de 2013, de <http://www.acs.com.hk/en/products/3/acr122u-usb-nfc-reader/>
- [2] Balaguera, Y. D. (2013). Agile methodologies in the development of applications.
- [3] Banes, M. (s.f.). *PR NewsWire*. Recuperado el 1 de Mayo de 2014, de PR NewsWire: <http://www.prnewswire.co.uk/news-releases/welcome-se-une-a-nfc-forum-como-miembro-asociado-153567265.html>
- [4] CASEMaker Inc. (2000). What is Rapid Application Development (RAD)? *CASEMaker Totem* .
- [5] CNXSoft. (s.f.). *Near Field Communication (NFC) Introduction and Software Development*. Recuperado el 20 de Febrero de 2014, de <http://www.cnx-software.com/>
- [6] Finkenzeller, K. (2010). *RFID HANDBOOK*. Munich: WILEY.
- [7] Harvick, R. (2012). *Agile Architecture for Service Oriented Component Driven Enterprises*. DataThunder.
- [8] Inside Secure. (2012). *Open NFC - NFC HAL Protocol*.
- [9] Martin, J. (1990). *Rapid Application Development*. New York: MacMillan Publishing Co.
- [10] McConnell, S. (1996). *Rapid Development*. Microsoft Press.
- [11] Motorola. (s.f.). *Motorola Global*. Recuperado el 1 de Diciembre de 2013, de https://motorola-global-es-latam.custhelp.com/app/answers/prod_answer_detail/a_id/95886/p/30,6720,8696
- [12] Murch, R. (2010). *Rapid Application Development Lifecycle*.
- [13] NFC Forum. (2010). NFC Digital Protocol. *NFC Forum* , 194.
- [14] Poslad, S. (2009). *Ubiquitous Computing*. Londres: WILEY.
- [15] Reilly, J., & Carmel, E. (1996). Does RAD live up to the hype? *IEEE Software* .
- [16] Ritu, A., Jayesh, P., & Mohan, T. (s.f.). Risks of Rapid Application Development. *COMMUNICATIONS OF THE ACM* .
- [17] Roland, M. (2015). *Security Issues in Mobile NFC Devices (T-Labs Series in Telecommunication Services)*. Springer.

- [18] Tutorial Point. (s.f.). Software Development Life Cycle. *tutorial point* .
Estado Unidos.
- [19] Vedat, C., Karen, O., & Ozdenizci, B. (2013). *PROFESSIONAL NFC APPLICATION DEVELOPMENT*.
- [20] Zeinoun, I. J. (2005). The Rapid Application Development Process.
Cambridge Technology Enterprises, Inc.

GLOSARIO

- **RAD:** Desarrollo rápido de aplicaciones (Rapid ApplicationDevelopment), es una metodología de desarrollo de software.
- **SDLC:** Ciclo de vida de desarrollo de software (Software DevelopmentLifeCycle), es el proceso de creación o modificación de los sistemas, modelos y metodologías que la gente usa para desarrollar estos sistemas de software.
- **NFC:** Comunicación de campo cercano (Near Field Comunnication), es una tecnología de comunicación inalámbrica.
- **RFID:** Identificación por radio frecuencia (Radio FrequencyIdentification), es un sistema de almacenamiento y recuperación de datos remoto.
- **UWB:** Banda ultra ancha (Ultra Wide Band) es una tecnología de radio en el rango de las PAN.
- **Zigbee:** es el nombre de la especificación de un conjunto de protocolos de alto nivel de comunicación inalámbrica.
- **IEEE:** Instituto de ingeniería eléctrica y electrónica.
- **ISO:** Organización Internacional de Normalización.
- **TAG:** sistema de almacenamiento y recuperación de datos remoto.
- **P2P:** (peer to peer) red de dispositivos donde unos o algunos aspectos funcionan sin clientes o servidores fijos.
- **FeliCa:** sistema de tarjetas contactless inteligentes RFID.
- **Android:** sistema operativo diseñado inicialmente para dispositivos móviles basado en el núcleo de Linux.
- **AndroidBean:** tecnología que permite la transmisión de datos desde sistemas operativos Android a través de tecnología NFC.
- **CASE:** Ingeniería de Software Asistida por Computadora) son diversas aplicaciones destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero.

- **JavaFX:** es una familia de productos y tecnologías empleadas para la creación de aplicaciones web que tienen las características y capacidades de aplicaciones de escritorio, incluyendo aplicaciones multimedia interactivas.
- **ERP:** son sistemas informáticos gerenciales que integran y manejan muchos de los negocios asociados con las operaciones de producción y de los aspectos de distribución de una compañía en la producción de bienes o servicios.
- **Dalvik:** máquina virtual que utiliza la plataforma para dispositivos móviles Android. Dalvik ha sido diseñada por Dan Bornstein con contribuciones de otros ingenieros de Google.
- **SNEP:** (Simple NDEF Exchange Protocol) protocolo que permite la transmisión de datos de una manera simple a través de NFC