



UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE COMPUTACIÓN
CENTRO DE ENSEÑANZA ASISTIDA POR COMPUTADOR - CENEAC

Aplicación Android para el intercambio de información entre dispositivos móviles con tecnología NFC integrada y etiquetas NFC

**Trabajo Especial de Grado presentado ante la
Ilustre Universidad Central de Venezuela por el
Br. Rafael Alejandro Machado Morales (C.I.V-20491784)
para optar al título de Licenciado en Computación.**

Tutora: Profa. Yusneyi Y. Carballo Barrera

Ciudad Universitaria de Caracas, 14 de Mayo de 2015

UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE COMPUTACIÓN
CENTRO DE ENSEÑANZA ASISTIDA POR COMPUTADOR - CENEAC

ACTA

Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación, para examinar el Trabajo Especial de Grado titulado **“Aplicación Android para el intercambio de información entre dispositivos móviles con tecnología NFC integrada y etiquetas NFC”** y presentado por el Br. **Rafael Alejandro Machado Morales (C.I.V-20491784)**, a los fines de optar al título de **Licenciado en Computación**, dejamos constancia de lo siguiente:
Leído como fue dicho trabajo, por cada uno de los miembros del jurado, se fijó el día ____ de _____ de _____, a las _____ horas, para que el autor lo defendiera en forma pública, lo cual éste hizo en _____ de la Escuela de Computación, mediante una presentación oral de su contenido, luego de lo cual respondió a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobar con la nota de ____ puntos.

En fe de lo cual se levanta la presente Acta, en Caracas el día ____ de _____ de _____.

Profa. Yusneyi Carballo Barrera
(Tutora)

Prof. Antonio Leal
(Jurado)

Prof. Néstor Méndez
(Jurado)

A Dios por darme sabiduría para lograr cada una de mis metas.
A mis padres Carmen y Pedro, y mis hermanos María Alejandra y Jorge Luis, por
su apoyo invaluable en cada uno de los momentos importantes de mi vida.
A mi tutora, Profa. Yusneyi Carballo Barrera por confiar en mí y en el proyecto
y compartir sus conocimientos para lograr un producto de calidad.
A mi familia extendida, mis amigos, Bonifacio, Emily, José Vicente, Laura, Pascual y
otros muchos compañeros de vida y estudios
por aportar ideas y buenos recuerdos.

A los que creen.

Gracias totales.



Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación
CENTRO DE ENSEÑANZA ASISTIDA POR COMPUTADOR - CENEAC

Aplicación Android para el intercambio de información entre dispositivos móviles con tecnología NFC integrada y etiquetas NFC

Autor: Rafael A. Machado Morales
C.I.: 20.491.784

Correo-e: rafael.machado@ciens.ucv.ve

Tutora: Profa. Yusneyi Carballo Barrera

Correo-e: yusneyi.carballo@ciens.ucv.ve

Fecha: 14 de Mayo de 2015

RESUMEN

La Comunicación de Campo Cercano o *Near Field Communication* (**NFC**) es una tecnología de comunicación de corto alcance, que permite el intercambio de datos entre dos dispositivos de manera inalámbrica. Los avances en el desarrollo de dispositivos móviles han permitido incluir los componentes necesarios para utilizar NFC en equipos de uso diario como tabletas, teléfonos y/o relojes inteligentes en una amplia gama de aplicaciones. Este trabajo describe el proceso de diseño e implementación de una aplicación para el Sistema Operativo Android que permite el intercambio de información entre dispositivos móviles con tecnología NFC integrada y etiquetas NFC. Asimismo se incluye una investigación que abarca conceptos generales de la tecnología NFC, el sistema operativo Android y el desarrollo de aplicaciones móviles, en la cual se basa la elección del enfoque utilizado para dar solución al problema planteado; además de la descripción de la metodología de desarrollo aplicada. Los resultados obtenidos durante la fase de pruebas indican que la aplicación cumple con los requerimientos solicitados, permitiendo el intercambio de una gran variedad de información entre dispositivos Android y etiquetas NFC, otorgando una amplia gama de usos en función de las necesidades del usuario.

Palabras Clave: *Near Field Communication* (NFC), Protocolo Comunicación de Campo Cercano, etiqueta NFC, Aplicación móvil Android,

Índice general

INTRODUCCIÓN	V
CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA	- 1 -
I.1 PROBLEMA DE INVESTIGACIÓN	- 1 -
I.3 OBJETIVO GENERAL DEL TRABAJO ESPECIAL DE GRADO	- 2 -
I.4 OBJETIVOS ESPECÍFICOS	- 2 -
I.5 ALCANCE	- 3 -
I.6 LIMITACIONES	- 3 -
CAPÍTULO II: MARCO TEÓRICO	- 4 -
II.1 TECNOLOGÍAS PARA COMUNICACIONES EN DISPOSITIVOS MÓVILES	- 4 -
II.1.1 Tecnología de Identificación por Radiofrecuencia o Radio Frequency Identification (RFID)	- 4 -
II.1.2 Tecnología de Tarjetas Inteligentes o Smart Cards	- 8 -
II.1.3 Evolución hacia la NFC	- 11 -
II.1.4 Tecnología NFC	- 12 -
II.2 SISTEMA OPERATIVO ANDROID	- 27 -
II.2.1 Características principales del sistema operativo Android	- 27 -
II.2.2 Componentes principales del sistema operativo Android	- 28 -
II.2.3 Dispositivos móviles con sistema operativo Android equipados con tecnología NFC	- 29 -
II.3 APLICACIONES MÓVILES	- 30 -
II.3.1 Definición de aplicación móvil	- 30 -
II.3.2 Arquitectura de las aplicaciones móviles	- 30 -
II.3.3 Clasificación de las aplicaciones móviles	- 31 -
II.3.4 Desarrollo de aplicaciones móviles para el sistema operativo Android	- 32 -
II.3.5 API de Android para NFC	- 34 -
II.4 ANTECEDENTES DE APLICACIONES PARA ANDROID QUE UTILIZAN CARACTERÍSTICAS NFC	- 36 -
II.5 CONCLUSIONES DEL CAPITULO	- 39 -
CAPÍTULO III: MARCO METODOLÓGICO	- 40 -
III.1 DESARROLLO INCREMENTAL E ITERATIVO	- 40 -
III.1.1 Etapas del ciclo de vida del Desarrollo Incremental e Iterativo	- 42 -
III.1.2 Beneficios del Desarrollo Incremental e Iterativo	- 43 -
III.1.3 Restricciones del Desarrollo Incremental e Iterativo	- 43 -
III.1.4 Artefactos del Desarrollo Incremental e Iterativo	- 44 -
III.1.5 Justificación	- 44 -
III.2 METODOLOGÍA A SEGUIR PARA EL DESARROLLO DE LA SOLUCIÓN	- 45 -
II.2.1 Artefactos	- 45 -
II.2.2 Justificación	- 47 -
CAPÍTULO IV: DESARROLLO DE LA APLICACIÓN	- 48 -
IV.1 OBJETIVO GENERAL DE LA APLICACIÓN	- 48 -
IV.2 OBJETIVOS ESPECÍFICOS DE LA APLICACIÓN	- 48 -
IV.3 ALCANCE DE LA APLICACIÓN	- 48 -
IV.4 PLATAFORMA	- 49 -
IV.5 DESCRIPCIÓN DE LA APLICACIÓN	- 51 -
IV.5.1 Usuarios a los que está dirigida	- 51 -
IV.5.2 Principales requerimientos funcionales	- 51 -
IV.5.3 Principales requerimientos no funcionales	- 52 -
IV.6 DESARROLLO DE LA APLICACIÓN	- 52 -

IV.6.1 Funcionalidades desarrolladas.....	- 52 -
IV.6.2 Diagramas de Casos de Uso	- 55 -
IV.6.3 Arquitectura de la aplicación.....	- 57 -
IV.6.4 Implementación de las funcionalidades	- 59 -
IV.6.5 Principales interfaces de la aplicación	- 70 -
IV.6.6 Pruebas de la aplicación.....	- 81 -
RESULTADOS DEL TRABAJO ESPECIAL DE GRADO.....	- 84 -
CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS	- 86 -
REFERENCIAS BIBLIOGRÁFICAS Y DÍGITALES.....	- 88 -
ANEXO A: CÓDIGO QR.....	- 91 -
ANEXO B: CODIFICACIÓN HUFFMAN.....	- 92 -
ANEXO C: ENCUESTA UTILIZADA EN LA PRUEBA DE ACEPTACIÓN DE LA APLICACIÓN	- 94 -

Índice de figuras

FIGURA 2-1. ESQUEMA DEL FUNCIONAMIENTO DEL SISTEMA RFID.	- 5 -
FIGURA 2-2. EVOLUCIÓN DE LA TECNOLOGÍA NFC.	- 12 -
FIGURA 2-3. INTERFAZ DE USUARIO DE LA APLICACIÓN NFC TRIGGER.....	- 36 -
FIGURA 2-4. INTERFAZ DE USUARIO DE LA APLICACIÓN NFC TOOLS.	- 37 -
FIGURA 2-5. INTERFAZ DE USUARIO DE LA APLICACIÓN NFC TAGWRITER BY NXP.	- 38 -
FIGURA 3-1. CICLO DE VIDA DEL DESARROLLO INCREMENTAL.....	- 41 -
FIGURA 3-1. TABLERO DE REQUERIMIENTOS REALIZADO EN LA HERRAMIENTA ONLINE TRELLO.	- 46 -
FIGURA 4-1. DIAGRAMA DE CASOS DE USO, NIVEL 0.	- 55 -
FIGURA 4-2. DIAGRAMA DE CASOS DE USO, NIVEL 1.	- 56 -
FIGURA 4-3. DIAGRAMA ENTIDAD – RELACIÓN DEL MODELO DE DATOS DE LA APLICACIÓN.	- 58 -
FIGURA 4-4. DIAGRAMA DEL MODELO RELACIONAL DE DATOS DE LA APLICACIÓN.....	- 58 -
FIGURA 4-5. DIAGRAMA DE SECUENCIA DE LA FUNCIONALIDAD “LECTURA DEL CONTENIDO ALMACENADO EN UNA ETIQUETA NFC”.	- 60 -
FIGURA 4-6. DIAGRAMA DE SECUENCIA DEL PROCESO DE ALMACENAMIENTO DE CONTENIDO EN LA APLICACIÓN. ..	- 61 -
FIGURA 4-7. DIAGRAMA DE SECUENCIA DE LA FUNCIONALIDAD “ESCRITURA DEL CONTENIDO ALMACENADO EN UNA ETIQUETA NFC”.	- 63 -
FIGURA 4-8. DIAGRAMA DE SECUENCIA DE LA FUNCIONALIDAD “ELIMINAR CONTENIDO ALMACENADO EN LA APLICACIÓN”.	- 65 -
FIGURA 4-9. DIAGRAMA DE SECUENCIA DE LA FUNCIONALIDAD “FILTRAR EL CONTENIDO ALMACENADO EN LA APLICACIÓN”.	- 67 -
FIGURA 4-10. INTERFAZ PRINCIPAL DE LA APLICACIÓN.	- 70 -
FIGURA 4-11. INTERFAZ PRINCIPAL DEL MÓDULO DE LECTURA.	- 71 -
FIGURA 4-12. INTERFAZ DE DESPLIEGUE DEL CONTENIDO LEÍDO.....	- 71 -
FIGURA 4-12.B. INTERFAZ DE DESPLIEGUE DEL CONTENIDO LEÍDO.....	- 72 -
FIGURA 4-13. INTERFAZ DE DESPLIEGUE DEL RESULTADO DE ALMACENAMIENTO DE CONTENIDO.	- 73 -
FIGURA 4-14. INTERFAZ PRINCIPAL DEL MÓDULO DE ESCRITURA DE CONTENIDO.....	- 74 -
FIGURA 4-15. INTERFAZ DE CREACIÓN Y EDICIÓN DE CONTENIDO.	- 75 -
FIGURA 4-16. INTERFAZ DE TRANSFERENCIA DE CONTENIDO.	- 76 -
FIGURA 4-17. INTERFAZ PRINCIPAL DEL MÓDULO DE ESCRITURA DE CONTENIDO.....	- 77 -
FIGURA 4-18. INTERFACES CON FUNCIONALIDADES PARA EL CONTENIDO.	- 78 -
FIGURA 4-19. INTERFACES DE LAS OPCIONES DE FILTRADO.	- 79 -

FIGURA 4-20. INTERFACES DE FUNCIONALIDADES EXTRAS.	- 80 -
FIGURA A-1. CÓDIGO QR GENERADO A PARTIR DEL URL HTTP://WWW.CIENS.UCV.VE	- 91 -
FIGURA B-1. ÁRBOL BINARIO GENERADO A PARTIR DE LA FRECUENCIA EXACTA DE LOS CARACTERES EN LA FRASE "THIS IS AN EXAMPLE OF A HUFFMAN TREE"	- 93 -
FIGURA C-1. FRAGMENTO DE LA ENCUESTA UTILIZADA DURANTE LAS PRUEBAS DE ACEPTACIÓN.	- 94 -
FIGURA C-1. GRÁFICA DE CUANTIFICACIÓN DE RESPUESTAS DE LA PREGUNTA N° 1.	- 95 -
FIGURA C-2. GRÁFICA DE CUANTIFICACIÓN DE RESPUESTAS DE LA PREGUNTA N° 2.	- 95 -
FIGURA C-3. GRÁFICA DE CUANTIFICACIÓN DE RESPUESTAS DE LA PREGUNTA N° 3.	- 96 -
FIGURA C-4. GRÁFICA DE CUANTIFICACIÓN DE RESPUESTAS DE LA PREGUNTA N° 4.	- 96 -
FIGURA C-5. GRÁFICA DE CUANTIFICACIÓN DE RESPUESTAS DE LA PREGUNTA N° 5.	- 97 -
FIGURA C-6. GRÁFICA DE CUANTIFICACIÓN DE RESPUESTAS DE LA PREGUNTA N° 6.	- 97 -

Índice de tablas

TABLA 2-1. COMPARACIÓN DE LOS TIPOS DE ETIQUETAS NFC.	- 21 -
--	--------

Introducción

La necesidad de los seres humanos por comunicarse ha sido evidente desde los inicios de la civilización y a la par de los avances tecnológicos surgen nuevas técnicas y/o herramientas que buscan facilitar y agilizar la transferencia de información. En la actualidad el uso de teléfonos celulares se encuentra ampliamente extendido en nuestra sociedad, y desde hace un par de décadas la comunicación por voz ha cedido terreno a nuevas tecnologías que permiten transferir algo más que palabras mediante el envío de mensajes simples de texto o multimedia. Muchos dispositivos móviles recientes ofrecen la posibilidad de enviar archivos de texto, imágenes, videos, coordenadas geográficas, enlaces a recursos web y muchos otros formatos de archivo mediante diversas tecnologías entre las que destacan las conexiones inalámbricas a Internet, el uso de *Wi-Fi* y *Bluetooth*, los sensores infrarrojos y la comunicación de campo cercano o NFC por sus siglas en inglés. Esta última, en particular, ha sido aprovechada por los desarrolladores del sistema operativo Android para explotar las posibilidades en los dispositivos equipados con esta tecnología, permitiendo, por ejemplo, utilizar un teléfono celular para emular una tarjeta de crédito para realizar compras o aplicar configuraciones a un teléfono o tableta con tan sólo acercarlo a una etiqueta NFC.

En este trabajo se presentarán los conceptos necesarios para analizar el origen, la evolución y las utilidades de la tecnología NFC, las principales características del sistema operativo Android y su relación con la NFC, así como también, enfoques a utilizar durante el desarrollo de aplicaciones para dicho sistema operativo que permitan compartir información entre dispositivos compatibles con esta tecnología. El resultado final de esta investigación es el desarrollo de una aplicación para Android capaz de compartir información mediante tecnología NFC.

La estructura del presente Trabajo Especial de Grado es la siguiente:

Capítulo I: Planteamiento del problema de investigación, donde se especifica el objetivo general y los objetivos específicos del Trabajo Especial de Grado, junto con el alcance, las preguntas de investigación y los resultados esperados.

Capítulo II: Marco teórico, el cual incluye la investigación de los conceptos relacionados a la tecnología NFC, el sistema operativo Android, la relación entre la tecnología NFC y Android

y el desarrollo de aplicaciones móviles que permitan obtener beneficio en dispositivos equipados con esta tecnología.

Capítulo III: Marco metodológico, donde se describe la metodología basada en el modelo incremental e iterativo utilizada para el desarrollo de la aplicación.

Capítulo IV: Desarrollo de la aplicación, en el que se detallan los objetivos generales y específicos de la aplicación, su alcance, la plataforma de implementación y se realiza una minuciosa descripción de la aplicación.

Posteriormente, se describen los Resultados del Trabajo Especial de Grado, y finalmente se ofrecen las Conclusiones y Recomendaciones del mismo, así como las Referencias Bibliográficas y digitales consultadas para su elaboración, junto con los anexos.

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA

En este capítulo se describe el problema cuya solución se busca alcanzar con este trabajo de investigación, el objetivo de la investigación, los objetivos específicos, el alcance del Trabajo Especial de Grado y las limitaciones identificadas durante la investigación.

1.1 Problema de investigación

Mucho han cambiado los teléfonos celulares desde que fueron presentados como dispositivos que pretendían facilitar las comunicaciones por voz entre las personas. En la actualidad existen versiones que incluyen antenas y sensores que permiten establecer comunicaciones con otros dispositivos electrónicos utilizando diversas tecnologías de comunicación inalámbrica como los sensores infrarrojos (IrDA, por sus siglas en inglés) y/o el *Bluetooth*, por mencionar apenas dos de las más conocidas.

En 2002, una tecnología conocida como NFC (*Near Field Communication*, por sus siglas en inglés) fue propuesta como una nueva forma de intercambiar un volumen moderado de datos de forma rápida y segura entre dos dispositivos compatibles, basándose en los conceptos de la RFID (*Radio Frequency Identification*, por sus siglas en inglés) y aportando mejoras ^[1]. Las características de la tecnología NFC incorporadas en los teléfonos móviles son utilizadas principalmente para compartir archivos de texto, información de contacto, recursos multimedia como pequeñas imágenes y videos cortos o simplemente un enlace a un recurso web, además de ser utilizadas para implementar sistemas de pago electrónico.

1.2 Justificación del Problema

ALS Telecom es una empresa cuyo principal producto es un conjunto de aplicaciones que facilitan el envío de grandes volúmenes de mensajes de texto utilizando principalmente la tecnología SMS con el fin de facilitar la comunicación interactiva entre las empresas y sus clientes. No obstante, la empresa también ofrece servicios de desarrollo de proyectos de información no relacionados con el envío de mensajes de texto y constantemente busca expandir su posición incursionado en nuevos mercados. Si bien la empresa ha participado en desarrollos Web para el consumo interno y externo de varios clientes, no ha estado involucrada en un desarrollo orientado al público. Con el interés de ampliar su catálogo de productos y estar a la vanguardia de la tecnología ha decidido desarrollar una aplicación para teléfonos

celulares con sistema operativo Android y con tecnología NFC integrada, con el fin de ofrecer a sus usuarios información oportuna en diversos escenarios de la vida cotidiana.

Conocer de forma rápida la ubicación actual y la de los lugares de interés al llegar a una nueva ciudad, tener acceso a la metadata e información relacionada de un libro y poder almacenar en un teléfono la posición exacta de donde se encuentra estacionado un auto son algunas de las situaciones en las que se evidencia la utilidad de la aplicación requerida por ALS TELECOM , dado que dicha información de interés puede ser almacenada en etiquetas con chips NFC incorporados y estar a disposición de aquellos usuarios que hayan instalado previamente la aplicación en sus dispositivos.

1.3 Objetivo general del Trabajo Especial de Grado

Desarrollar una aplicación para el sistema operativo Android que permita el intercambio de contenido entre dispositivos compatibles con la tecnología NFC y etiquetas NFC.

1.4 Objetivos específicos

1. Levantar los requerimientos de la aplicación para definir sus lineamientos.
2. Desarrollar un módulo que implemente el modo de operación Lectura de la tecnología NFC.
3. Desarrollar un módulo que implemente el modo de operación Escritura de la tecnología NFC.
4. Definir e implementar un modelo de datos que permita almacenar el contenido leído.
5. Desarrollar un sistema de gestión del contenido almacenado en la aplicación.
6. Realizar las pruebas pertinentes al proceso de intercambio de información y al sistema de gestión del contenido almacenado.
7. Documentar la implementación de las funcionalidades del sistema.

1.5 Alcance

El alcance propuesto contempla el diseño y desarrollo de una aplicación móvil para el sistema operativo Android que permita la lectura y escritura de contenidos desde y hacia etiquetas NFC, admitiendo además almacenar, clasificar y compartir el contenido recibido.

Para lograr este alcance, se abarcaron los siguientes elementos:

- Diseño e implementación de un módulo que permita la lectura del contenido almacenado en una etiqueta NFC.
- Diseño e implementación de un módulo que permita la lectura del contenido almacenado en una etiqueta NFC.
- Diseño e implementación de un modelo de datos que permita almacenar el contenido leído desde una etiqueta NFC, así como también permita clasificar el contenido y compartirlo con distintas aplicaciones del sistema operativo.

Con la estructura creada, la aplicación puede ser adaptada para generar y consumir contenido específico en función de los servicios que la empresa ALS Telecom desee ofrecer.

1.6 Limitaciones

Las limitaciones de la presente investigación son las siguientes:

- El contenido a compartir entre el dispositivo y las etiquetas NFC dependerá de la capacidad máxima permitida por cada etiqueta.
- La comunicación entre dispositivos celulares y etiquetas NFC está sujeta a la compatibilidad de los componentes de cada equipo indicada por los fabricantes de los mismos.

CAPÍTULO II: MARCO TEÓRICO

II.1 Tecnologías para comunicaciones en dispositivos móviles

Existe una extensa y variada gama de tecnologías de comunicación implementadas en los dispositivos móviles de uso diario como teléfonos, computadoras personales, relojes, etc. con el fin de facilitar el intercambio de información entre estos. A continuación se presentan los principales aspectos de tres tecnologías de comunicación inalámbrica: la Identificación por Radiofrecuencia o RFID, las Tarjetas Inteligentes o *Smart Cards* y la Comunicación de campo Cercano o NFC.

II.1.1 Tecnología de Identificación por Radiofrecuencia o Radio Frequency Identification (RFID)

Radio Frequency Identification o Identificación por Radiofrecuencia es una tecnología inalámbrica para intercambiar datos entre un lector RFID y una etiqueta RFID a través de ondas de radio. Su principal función es la identificación de determinados objetos a distancia utilizando las ondas de radio. Para conseguirlo, esta tecnología proporciona soporte para el almacenamiento en etiquetas o tarjetas RFID que contienen la información necesaria para el reconocimiento. ^{[1][2]}

II.1.1.1 Funcionamiento de la RFID

Básicamente consta de tres partes como se ven en la figura 2-1. Existe un lector RFID que de manera periódica busca en su zona de alcance la información contenida en las señales que son emitidas por una etiqueta o transpondedor RFID. ^[2] Estas etiquetas están compuestas por un circuito integrado o *Integrated Circuit* (IC, por sus siglas en inglés) y una antena; por lo general están adheridas a un producto, persona o animal con el propósito de identificarlo o rastrearlo. Una vez se hayan obtenidos los datos, el lector los transfiere a un subsistema de procesamiento para la interpretación y el tratamiento correspondiente.

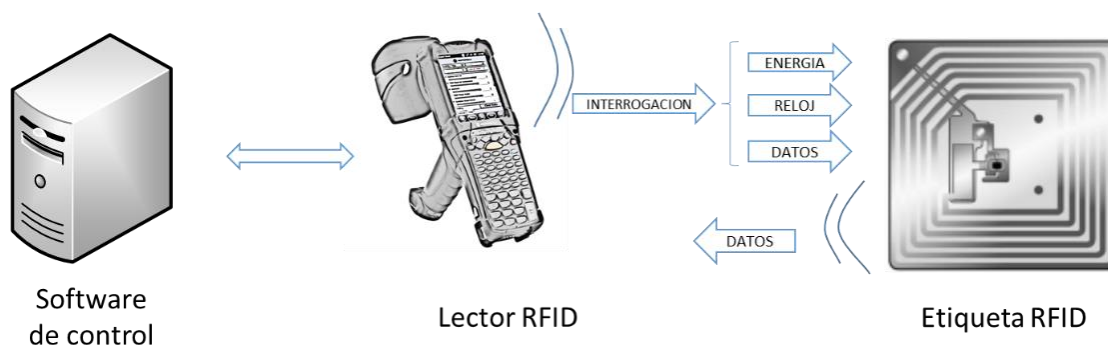


Figura 2-1. Esquema del funcionamiento del sistema RFID.
Fuente: Ortiz. S. Near Field Communication.

II.1.1.2 Clasificación de las etiquetas RFID

Los transpondedores o etiquetas RFID consisten en un elemento de apareamiento y un circuito integrado que se encarga de que la data sea transferida. Las etiquetas RFID tienen una alta capacidad para almacenar grandes cantidades de información. Pueden ser clasificadas en tres grandes tipos tomando en cuenta su fuente de alimentación:

- Etiquetas pasivas: No poseen fuente de alimentación propia, por lo que utilizan la pequeña corriente inducida por la energía que reciben del lector, misma que es suficiente para poner en marcha su circuito y para realizar la transmisión de datos. Son las más económicas.
- Etiquetas semi-pasivas: Reciben parte de su energía de una fuente propia de alimentación. Esta batería solo es suficiente para alimentar solo sus circuitos, para la transmisión de información utilizan la energía emitida por el lector.
- Etiquetas activas: Cuentan con su propia fuente de alimentación tanto para sus circuitos, como para la transmisión de información. En cuanto a costo son menos económicas que semi-pasivas pero poseen mayor capacidad de almacenamiento.

II.1.1.3 Usos comunes de RFID

Esta tecnología es usada alrededor del mundo para una amplia variedad de aplicaciones. A continuación se presentan algunos ejemplos:

- **Identificación y control**
 - **Sistemas de inventario:** El control de inventario y el rastreo de artículos son las principales áreas de uso de RFID. La tecnología RFID permite a las compañías realizar inventario de forma fácil y rápida, agilizando los procesos contables y optimizando el monitoreo de productos.
 - **Control de animales y humanos:** Es una de las aplicaciones del RFID más antiguas. Se utilizan chips pasivos incrustados en el animal a nivel subcutáneo que permite su identificación para operaciones en grandes haciendas y esas áreas donde el terreno es complicado y el rastreo de animales es difícil. Además, esta aplicación también es utilizada para el seguimiento de pacientes dentro de hospitales, donde se utiliza una pulsera que permite localizar al paciente del hospital y también permite acceder a su historial.
 - **Control de acceso:** Permite controlar el acceso de funcionarios a los edificios, además de registrar y gestionar los horarios.
- **Sistemas de pago:** RFID se presenta como una opción para sistemas de pago, como por ejemplo la compra de boleto en el autobús urbano. En Caracas, Venezuela se utilizan etiquetas RFID en tarjetas para almacenar el saldo o número de viajes de los usuarios del Sistema Metro de Caracas. Cada vez que el cliente pasa la tarjeta por uno de los lectores ubicados en los torniquetes de la estaciones del sistema, es descontado un viaje de su cuenta en la tarjeta. Sin embargo, un punto débil es que este sistema no ofrece medidas de seguridad ni autenticación de los pasajeros.

II.1.1.4 Ventajas de RFID

A continuación se listan algunas de las virtudes de la tecnología RFID. ^[3] ^[4]

- Es un sistema bastante robusto y posee un buen desempeño en ambientes severos (lugares húmedos, sucios, y con variaciones de temperatura).
- No es necesario el contacto visual entre la etiqueta y el lector para la transferencia de datos o la comunicación.
- Un lector RFID puede leer múltiples etiquetas en simultáneo.
- Las etiquetas RFID son resistentes y duraderas debido a que es posible insertarlas en materiales robustos.
- Con la tecnología RFID se puede brindar a un producto una identidad propia dentro de una línea de producción, la capacidad de comunicarse con su ambiente y la habilidad de conservar y obtener la información del mismo.

II.1.1.5 Problemas o desventajas de RFID

En las siguientes líneas, se mencionan los inconvenientes propios de este sistema ^[10] ^[11]:

- RFID no cuenta con un estándar internacional referente al uso de frecuencias, esto ocasiona problemas de incompatibilidad en sistemas utilizados entre algunos países y otros.
- El principal punto débil del RFID, es que no provee de mecanismos de seguridad para proteger la información, ante las amenazas de ataques a la privacidad.
- La capacidad de almacenamiento y el procesamiento en las etiquetas RFID son insuficientes para implementaciones de mecanismos de seguridad tales como técnicas criptográficas. Este hecho contribuye a que se complique el establecimiento de conexiones seguras para evitar que los datos sean captados y leídos por intrusos.
- Entre el lector y las etiquetas puede ocurrir una saturación de etiquetas, debido a la excesiva lectura de etiquetas al mismo tiempo, o la redundancia en la lectura, que se produce cuando una etiqueta es detectada por varios lectores.
- Otra posible interferencia en el sistema es lo que se conoce como escudo electromagnético, que es el efecto producido cuando un material conductor se encuentra posicionado entre una etiqueta y un lector, por ejemplo, envolver con aluminio una etiqueta RFID.

- Con respecto a la seguridad, otra falencia es que la etiqueta puede ser leída a cierta distancia sin que el usuario tenga conocimiento de este hecho.
- El costo de la implementación del RFID en comparación con el sistema de código de barras es el principal factor limitante para su instalación, aunque con los desarrollos y mejoras, el uso de la tecnología RFID será cada vez más económico.

II.1.2 Tecnología de Tarjetas Inteligentes o *Smart Cards*

El uso de las tarjetas inteligentes en diversas tareas cotidianas se ha expandido notablemente desde su introducción al mercado. En la presente sección se describen sus principales características y aplicaciones.

II.1.2.1 Definición de tarjeta inteligente

Una tarjeta inteligente o *Smart Card* es una ficha o rectángulo plástico que incluye un circuito integrado o *IC* (por sus siglas en inglés) que puede ser una unidad de memoria con o sin un microcontrolador seguro. Es una solución prometedora para almacenamiento eficiente, el procesamiento y la transferencia de datos así como para proveer un ambiente de multiplicación seguro.

II.1.2.2 Funcionamiento de una tarjeta inteligente

Un sistema típico de tarjetas inteligentes, contiene tarjetas inteligentes, lectores de tarjetas y un sistema de procesamiento o *back-end*. Las tarjetas se pueden comunicar con un lector utilizando contacto físico (caso de tarjetas inteligentes de contacto) o de forma remota a interfaces de Radiofrecuencia sin contacto (caso de tarjetas inteligentes sin contacto). El lector se conecta al sistema de *back-end*, el cual almacena, procesa y gestiona la información.

II.1.2.3 Clasificación de las tarjetas inteligentes

Las tarjetas inteligentes se pueden clasificar según su capacidad de procesamiento o según su mecanismo de comunicación. ^[1]

- Según la capacidad de procesamiento:

En términos de capacidades de procesamiento, las tarjetas inteligentes son divididas en dos grupos: Las basadas en memoria y las basadas en microprocesador.

- Tarjetas inteligentes basadas en memoria o *Memory-Based Smart Cards*: Este tipo de tarjetas pueden almacenar cualquier tipo de información ya sea financiera, personal u otro tipo de datos privados. Sin embargo, no tienen ningún tipo de capacidad de procesamiento. Necesitan comunicarse con un dispositivo externo como un lector de tarjetas usando protocolos síncronos para manipular la información almacenada en estas. Este tipo de tarjetas son ampliamente usadas, por ejemplo, como tarjetas telefónicas pre-pagadas.
- Tarjetas inteligentes basadas en microprocesadores o *Memory-Based Smart Cards*: Este tipo de tarjetas tienen capacidades de procesamiento de datos incluidas. Poseen tanto un microprocesador como una memoria. El microprocesador incluido gestiona la asignación de memoria y la gestión de los datos. Las tarjetas inteligentes basadas en microprocesadores son comparables con pequeñas computadoras sin fuente de poder interna. Estas tarjetas inteligentes poseen un sistema operativo u *Operative System* (OS, por sus siglas en inglés) llamado SCOS, por *Smart Card Operative System*, habilitando al usuario para administrar la información contenida en la tarjeta y permitiéndoles ser multifuncionales. Estos sistemas operativos pueden almacenar información y realizar complejos cálculos con los datos almacenados.

- Según su mecanismo de comunicación:

Las tarjetas inteligentes son divididas en tres grupos principales en términos de su mecanismo de comunicación con dispositivos externos: Tarjetas inteligentes de contacto, tarjetas inteligentes sin contacto y tarjetas inteligentes híbridas.

- Tarjetas inteligentes de contacto o *Contact Smart Cards*: Este tipo de tarjetas incluyen un micro módulo que contiene una tarjeta de circuito integrado de silicio que posee además memoria y un microprocesador. Esta tarjeta de circuito integrado es una lámina de contacto conductivo típicamente chapado en oro

ubicado en la superficie de la tarjeta inteligente. Un dispositivo externo provee una conexión eléctrica directa a la lámina de contacto conductivo cuando la tarjeta inteligente es insertada en él. La transmisión de comandos, datos e información de estado de la tarjeta tiene lugar sobre estos puntos de contacto físico. Estos dispositivos externos son usados como medios de comunicación entre las tarjetas inteligentes de contacto y una computadora anfitrión, la cual puede ser un terminal de puntos de servicio o *Point of Sale*, *POS* (por sus siglas en inglés) o dispositivos móviles.

- Tarjetas inteligentes sin contacto o *Contactless Smart Cards*: Es un tipo de tarjeta que es procesada sin la necesidad de un contacto físico con dispositivo externo. Es una combinación de un microchip embebido y una antena, la cual permite rastrear la tarjeta. Esta antena está formada por varios cables. En este tipo de tarjetas la información es almacenada en el microchip, el cual tiene un microcontrolador seguro y una memoria interna. A diferencia de las tarjetas inteligentes por contacto, la fuente de poder de las tarjetas inteligentes sin contacto es conseguida con su antena incorporada. Los campos electromagnéticos de la tarjeta proveen la energía necesaria para la transferencia de datos entre la tarjeta y un dispositivo externo.
- Tarjetas inteligentes híbridas: Se pueden encontrar modelos híbridos de tarjetas inteligentes como tarjetas de interfaz dual y tarjetas híbridas. Una tarjeta de interfaz dual las interfaces de las tarjetas por y sin contacto para acceder al mismo chip con un alto nivel de seguridad. Una tarjeta híbrida contiene dos chips. Uno de estos es utilizado por una interfaz de contacto y el otro es utilizado por una interfaz sin contacto. Estos chips son independientes.

II.1.2.4 Usos comunes de las tarjetas inteligentes

El primer uso de las tarjetas inteligentes fue como tarjetas telefónicas pre-pagadas implementadas en Europa a mediados de 1980. Eran realmente simples tarjetas de memoria. Posteriormente, las áreas de aplicación incrementaron vastamente y en la actualidad algunos de los principales ámbitos de aplicación para las tarjetas inteligentes basadas en microprocesadores son finanzas, comunicaciones, identificación, control de acceso físico,

transporte, lealtad y cuidado de la salud. Incluso, una misma tarjeta puede tener varias aplicaciones. ^[1]

II.1.3 Evolución hacia la NFC

La Comunicación de Campo Cercano o *Near Field Communication* NFC brinda nuevas funciones a la tecnología RFID. Esto se debe a la combinación de un lector y una etiqueta RFID en un mismo equipo NFC. De esta manera se facilita la comunicación en ambas direcciones entre dos dispositivos y se elimina la separación funcional de los esquemas RFID, es decir, el lector por un lado y las etiquetas por el otro. ^[5]

Es necesario considerar que la tecnología NFC tiene algunas diferencias con RFID, a pesar de que ambas utilizan el mismo tipo de chip y que la primera deriva de la segunda. Lo que debe quedar en claro es que aunque NFC es un nuevo estándar que aparece a raíz de RFID, su misión no es reemplazarlo, sino más bien, complementarlo aumentando sus funciones. ^[6]

Realizando una comparación entre NFC y RFID, encontramos que la primera posee la capacidad de cómputo necesaria para ejecutar operaciones, hecho que hace fácil su integración en dispositivos como teléfonos móviles. ^[6] Además NFC provee el modo de comunicación *Peer-to-Peer* (P2P, por sus siglas en inglés), en cual permite intercambiar información entre dos dispositivos. ^[7] NFC integra las tecnologías de RFID y la de las tarjetas inteligentes sin contacto dentro de los teléfonos móviles. ^[1]

Otro aspecto que tiene NFC es que permite rangos pequeños de comunicación, y por ende posee una seguridad inherente, punto que lo hace preferible para temas de comunicación que requieren seguridad como los medios de pago o intercambio de información personal. Además NFC no puede ser activado de forma remota, involuntariamente o por accidente. El teléfono obliga a que deba existir un acercamiento entre dispositivos antes de iniciar una comunicación. ^[7]

Con todo esto, cabe mencionar que se hace necesaria la evolución hacia el NFC para aprovechar la robusta seguridad, la versatilidad por el hecho de que puede utilizarse en dispositivos móviles, la facilidad en el uso y la posibilidad de servir como una tecnología que brinde al usuario información útil y le permita interactuar con su ambiente.

II.1.4 Tecnología NFC

Derivada de las dos tecnologías revisadas anteriormente, la Comunicación de Campo Cercano o *Near Fiel Communication* NFC representa una alternativa que incluye los beneficios de sus predecesoras y aporta importantes mejoras de seguridad y facilidad de uso.

II.1.4.1 Definición de NFC

Near Field Communication (NFC), es una tecnología de comunicación de corto alcance, que permite el intercambio de datos entre dos dispositivos de manera inalámbrica. Es compatible con infraestructuras RFID, dado a que es un derivado del mismo. ^[5] La evolución de la tecnología NFC es ilustrada en la figura 2-2. Las áreas grises en la figura indican los desarrollos tecnológicos que soporta el ambiente NFC directamente.

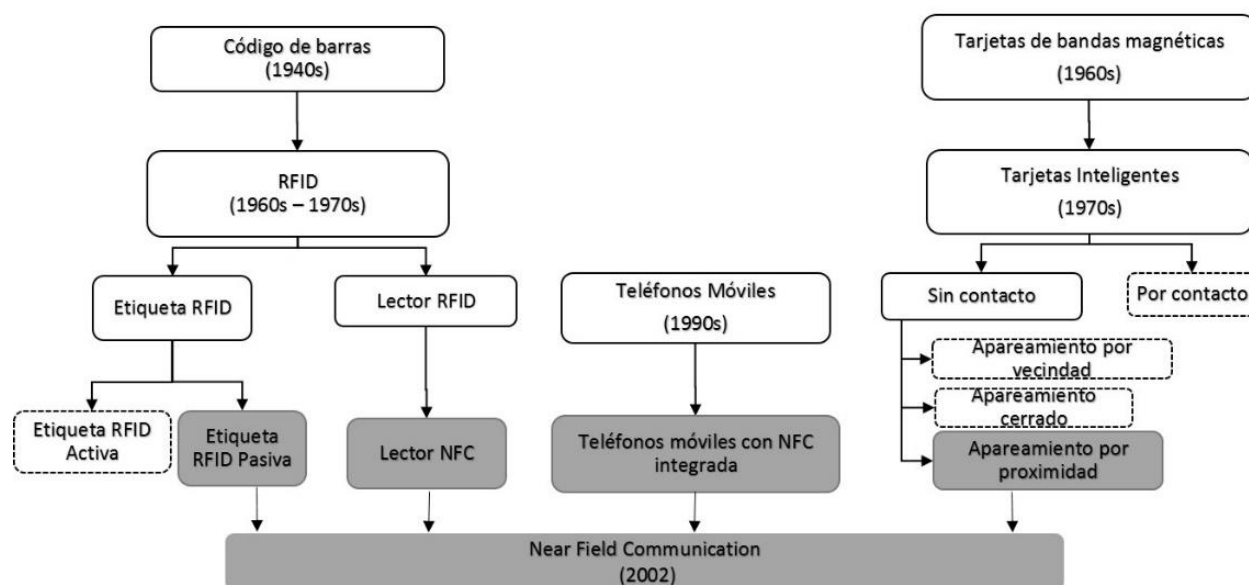


Figura 2-2. Evolución de la tecnología NFC.

Fuente: Coskun V., Ok K., Ozdenizci B. (2013). *Professional: NFC Application Development for Android™*. ^{1**}

^{1**} Traducido y diagramado por el autor.

La NFC fue presentada conjuntamente por Philips y Sony a mediados de 2002, siendo adoptada por la Asociación de Manufacturadores de Computadores Europea (ECMA, por sus siglas en inglés) a finales de ese mismo año. La Organización Internacional para la Estandarización (ISO, por sus siglas en inglés) y la Comisión Electrotécnica Internacional (IEC, por sus siglas en inglés) adoptaron la tecnología NFC en Diciembre de 2013. En 2004, Nokia, Philips y Sony fundaron el *NFC Forum* para promocionar la tecnología NFC y sus servicios.

II.1.4.2 NFC Forum

En 2004 se constituye el *NFC Forum*, definiéndose como una organización sin fines de lucro que busca promocionar la utilización de NFC en dispositivos a través del desarrollo de ciertas especificaciones que intentan unificar los sistemas NFC. ^[8]

Además de sus creadores, Philips, Sony y Nokia, el *NFC Forum* cuenta con más de una centena de miembros, entre ellos, empresas del rubro tecnológico, organismos económicos y otras organizaciones sin fines de lucro.^[9]

Esta asociación fomenta el desarrollo de un ambiente en donde se consigan que las aplicaciones para el NFC sean seguras y puedan funcionar unas con otras sin problema. Para lograr esto se definen especificaciones tanto para la arquitectura de los sistemas, como los protocolos para lograr la operatividad en conjunto.

II.1.4.3 Características de la tecnología NFC

La NFC proporciona comunicación inalámbrica de corto alcance mediante un campo magnético que permite el intercambio de datos y opera en un ambiente en donde la distancia máxima entre dispositivos es de 20cm. El sistema se maneja en la banda de frecuencia no licenciada de 13.56 MHz, y un ancho de banda que oscila 7 KHz a cada lado de la frecuencia. Las comunicaciones pueden ser half o full dúplex. Se utiliza el esquema de modulación Amplitud Shift Keying (ASK) y la codificación Manchester. La velocidad de transmisión puede ser de 106, 212 o 404 kbps, y es fijada por el dispositivo que inicia la conexión. ^[9]

II.1.4.4 Funcionamiento de la tecnología NFC

Un dispositivo NFC que comienza la comunicación y controla el intercambio de información es conocido como iniciador (similar al lector RFID), y el que responde al iniciador es conocido como objetivo. La comunicación puede realizarse en modo activo o en modo pasivo. ^[10] El sistema de corto alcance se compone de dos elementos: un iniciador y un objetivo, en donde cualquier dispositivo con NFC puede adoptar las funciones o el comportamiento de una de estas partes. El NFC puede ser instalado en cámaras fotográficas, reproductores, televisores, teléfonos celulares y hasta en controles remotos. ^[5]

En los sistemas NFC se pueden distinguir cinco etapas importantes que están presentes durante el establecimiento de la conexión. Estas fases son: ^[11]

1. Descubrimiento: En esta fase inicial, los dispositivos se rastrean mutuamente y luego inician el reconocimiento.
2. Autenticación: Cada uno de los dispositivos verifica si en el otro extremo su par está autorizado o si deben establecer una conexión segura a través de un cifrado correspondiente.
3. Negociación: Hasta este punto se definen parámetros como la tasa de transmisión, la identidad del dispositivo, la aplicación, y si es el caso, también la acción que van a solicitar.
4. Transferencia: En esta fase ya puede realizarse el intercambio de datos.
5. Confirmación: El receptor confirma el establecimiento de la comunicación y la transferencia de los datos.

II.1.4.5 Estandarización de la tecnología NFC

Existen diferentes estándares para el NFC, como los definidos por ISO/IEC, el ETSI (*European Telecommunications Standards Institute*) y también la ECMA. Estos especifican aspectos importantes en los sistemas NFC como la tasa de transferencia el esquema para la codificación y modulación así como otros parámetros. ^[8]

Entre los protocolos más importantes destacan:

- ECMA-340 (ISO/IEC 18092): define la interfaz y modo de operación *Peer-to-Peer* conocido como NFCIP-1, por *NFC Interface and Protocol-1*.
- ECMA- 352 (ISO/IEC 21481): define una segunda versión de la interfaz y modo de operación en NFC (NFCIP-2).
- ISO/IEC 14443: define aspectos de la proximidad en tarjetas inteligentes sin contacto y sus características.

II.1.4.6 Modos de Comunicación de la tecnología NFC

Como se mencionó anteriormente, para establecer una comunicación existen dos modos:

- Modo pasivo: En este modo es el iniciador el que dispone de fuente eléctrica propia para funcionar, y debe generar una señal para el intercambio de datos. Por el otro lado el dispositivo receptor u objetivo debe aprovechar el campo incidente del iniciador para el funcionamiento de sus circuitos.
- Modo Activo: En este modo ambos dispositivos, tanto el iniciador como el objetivo poseen energía propia, por lo que son capaces de generar el campo electromagnético para la transferencia de datos.

De igual forma, el dispositivo que posee la fuente de energía es conocido como dispositivo activo, y el que no la posee como dispositivo pasivo.

Un iniciador necesita siempre ser un dispositivo activo porque requiere una fuente de energía para iniciar la comunicación. El objetivo, no obstante, puede ser un dispositivo activo o uno pasivo. Si el objetivo es un dispositivo activo, este usa su propia fuente de energía para responder; si es un dispositivo pasivo, usa la energía creada por el campo electromagnético generado por el iniciador.

II.1.4.7 Modos de Operación de la tecnología NFC

Para los dispositivos NFC es posible establecer comunicación con el otro par actuando como etiqueta o como lector/escritor. El *NFC Forum* define los siguientes modos de operación: *Peer-*

to-Peer o Entre pares, Lector/Escritor (*Reader/Writer*) y Emulación de tarjeta inteligente (*Smart Card emulation*). ^[1]

- Modo Lector/Escritor: este modo trata la comunicación de un dispositivo con tecnología NFC integrada, como un teléfono móvil, con una etiqueta NFC con el objetivo de leer información desde la etiqueta o almacenar datos en la misma.

En el modo lector, el iniciador lee datos desde una etiqueta NFC. Además de contener la información solicitada, la etiqueta también posee el programa que envía dicha información al iniciador.

En el modo escritor, el teléfono móvil actúa como el iniciador y escribe datos en la etiqueta. Si la etiqueta ya contiene alguna información previa al proceso de escritura, esta es sobrescrita. El algoritmo puede ser diseñado para que el iniciador actualice modificando la información existente en lugar de sobrescribirla.

Un teléfono móvil puede realizar varias acciones después una vez haya leído los datos desde la etiqueta. Si la etiqueta almacena un URL, el teléfono puede lanzar un navegador web y mostrar la página web recibida, por ejemplo. Las características de los teléfonos móviles, como el poder de procesamiento, las capacidades de audio y video, y el acceso a Internet proveen muchas oportunidades tanto como para los usuarios como para los proveedores de servicio cuando el modo lector/escritor es utilizado.

A continuación se describe el flujo básico de este modo de operación.

- Modelo de uso genérico del modo lector/escritor:
 - Solicitud de lectura: un usuario solicita la información al poner en contacto un teléfono móvil con una etiqueta NFC, la cual puede estar embebida en distintos componentes como el paquete de un producto.
 - Transferencia de datos: los datos que residen en la etiqueta son transferidos al teléfono móvil.

- Procesamiento dentro del dispositivo: cuando la información es transferida al teléfono móvil, esta puede ser usada para muchos propósitos, como lanzar una aplicación, mostrar información al usuario, etc.
 - Uso adicional del servicio: este paso adicional toma ventaja de las capacidades del teléfono móvil y la a menudo involucra la capacidad de conectarse a Internet. Cuando la información es procesada en el teléfono móvil, esta puede ser usada para otras operaciones vía Internet.
 - Solicitud de escritura: el usuario solicita escribir datos en la etiqueta haciendo contacto entre esta y el teléfono móvil.
 - Reconocimiento: la etiqueta NFC responde con el *acknowledgment data* o ACK, informando al usuario sobre el éxito de la operación.
- Modo *Peer-to-Peer*: este modo habilita dos dispositivos móviles con tecnología NFC integrada para intercambiar información como un registro de contacto, un mensaje de texto, o cualquier otro tipo de dato. Este modo tiene dos opciones estandarizadas: NFCP-1 y *Logical Link Control Protocol* (LLCP o Protocolo de control de enlace lógico) el cual es utilizado en el tope del NFCIP-1.

Este último protocolo toma ventaja del paradigma iniciador-objetivo en el cual el iniciador y el objetivo son definidos antes de iniciar la comunicación. Sin embargo, los dispositivos son idénticos, en la comunicación a nivel del LLCP. Después del *handshake* o reconocimiento inicial, la aplicación que se ejecuta en la capa de aplicación toma la decisión. Debido a la energía embebida en los teléfonos móviles, ambos dispositivos están en modo activo durante la comunicación en modo *Peer-to-Peer*. La información es enviada sobre un canal bidireccional *half-duplex*, es decir, cuando un dispositivo está transmitiendo, el otro tiene que escuchar y debería iniciar la transferencia de datos cuando el primero termine.

A continuación se describe el flujo básico de este modo de operación.

- Modelo de uso genérico del modo *Peer-to-Peer*:
 - Solicitud y transferencia de datos: dos usuarios intercambian datos a través de sus teléfonos móviles.

- Uso adicional del servicio: cuando la información es compartida entre los teléfonos móviles, esta puede ser utilizada opcionalmente para propósitos adicionales como almacenar una tarjeta de contacto.
- Modo emulación de tarjeta inteligente: este modo otorga la oportunidad a un dispositivo móvil con tecnología NFC integrada de funcionar como una tarjeta inteligente sin contacto. Un dispositivo móvil puede almacenar múltiples aplicaciones compatibles con esta tecnología. Los casos de uso mayormente implementados de la emulación de tarjetas de este tipo son tarjetas de crédito, tarjetas de débito y tarjetas de lealtad.

En este modo de operación, un teléfono móvil con NFC no genera su propio campo de Radiofrecuencia, sino es el lector NFC quien crea el campo. Este comportamiento es atípico ya que el teléfono móvil es un dispositivo activo.

La emulación de tarjetas inteligentes sin contacto permite aplicaciones de pago y boletería, además es compatible con la infraestructura existente de tarjetas inteligentes.

A continuación se describe el flujo básico de este modo de operación:

- Modelo de uso genérico del modo emulación de tarjeta inteligente sin contacto:
 - Solicitud de servicio: El usuario hace una solicitud a un proveedor de servicio al poner en contacto un teléfono móvil y un lector NFC. La información solicitada es transferida desde teléfono móvil, hacia el proveedor servicio a través del lector NFC.
 - Servicios de fondo: El proveedor de servicios ejecuta los servicios de *back-end* solicitados después de obtener la información solicitada desde el teléfono móvil del usuario. Ejemplos de estos servicios son la autorización de las tarjetas de crédito y validaciones de tickets.
 - Uso del servicio + información: En este paso opcional, el proveedor de servicio retorna un servicio al usuario, como por ejemplo la autorización de un pago.

II.1.4.8 Arquitectura NFC en dispositivos

En un dispositivo móvil NFC es posible distinguir entre dos componentes fundamentales:

[6]

- Chip NFC y antena: Este conjunto permite la comunicación y el intercambio de datos entre los sistemas NFC a corta distancia, por medio de un campo magnético. El chip se encuentra conectado al controlador banda base del teléfono, que es el encargado de la comunicación del móvil.
- Elemento seguro o *Secure Element* (SE): Es un chip independiente que contiene las aplicaciones basadas en claves de seguridad y tiene como propósito permitir las transacciones seguras. Existen varias implementaciones para este elemento y varían según la ubicación del mismo dentro del móvil.

A continuación se detallan las implementaciones para el elemento seguro: [7]

- SE incorporado en el circuito del móvil: es la arquitectura más utilizada en los proyectos a nivel mundial. En este caso el SE puede ser un chip ya montado en la placa base o conectado a ella de alguna manera. Su principal ventaja es que ya posee todas las certificaciones de hardware y software necesarios, sin embargo, este modelo acarrea un gran problema cuando el usuario quiera cambiar de teléfono y deba de gestionar las credenciales de pago.
- Tarjeta de memoria utilizada como SE: en esta implementación una tarjeta de memoria incorpora un chip con un microcontrolador y una memoria flash.
- Tarjeta SIM como SE: esta solución es más llamativa para las operadoras, porque de esta forma toda la gestión de la información estaría a su cargo, utilizando la tecnología que hoy en día es empleada para configurar los dispositivos de forma remota.

II.1.4.9 Etiquetas NFC

Las etiquetas NFC son pequeños dispositivos con un chip NFC integrado y una unidad de memoria. Son un tipo de dispositivo pasivo, es decir, no poseen una fuente de energía propia por lo que utilizan la energía generada por el iniciador. Pueden ser adhesivas o venir dentro de chapas, llaveros plásticos o pulseras.

El *NFC Forum* ha definido las especificaciones de 4 tipos de etiquetas indicando sus características de tal manera que se logre una compatibilidad y operatividad entre los dispositivos en sus diferentes modos de lectura o escritura.

A continuación se describen las principales características de cada tipo de etiqueta NFC.

- Etiqueta tipo 1: este tipo de etiqueta está basado en el estándar ISO/IEC 14443-A. Pueden ser leídas y escritas. Los usuarios pueden modificar su contenido y configurarlas como de sólo lectura cuando sea necesario. La disponibilidad de memoria es de hasta casi 1 kb (96 Kbyte) pudiendo ser expandida hasta 2 Kbyte, la cual sólo es suficiente para almacenar la URL de una página web o una cantidad de datos similar. Como resultado de su simplicidad, esta etiqueta tiene un costo accesible y aún puede ser utilizada en la mayoría de las aplicaciones NFC.
- Etiqueta tipo 2: estas comparten la mayoría de las características de las etiquetas tipo 1. La mayor diferencia entre estos dos tipos es que el tamaño mínimo de la memoria es de 48 Kbyte pudiendo igualmente estar equipada con hasta 2 Kbyte de memoria.
- Etiqueta tipo 3: las etiquetas de este tipo están basadas en el estándar industrial japonés (JIX, por sus siglas en inglés) X 6319-4, también conocido como FeliCa. Estas etiquetas son pre configuradas por los fabricantes para ser tanto leídas como re-escritas, o sólo en modo lectura. Su capacidad de memoria es variable, disponiendo de una memoria teórica máxima de 1 Mbyte por servicio. Este tipo es más costoso comparado con los dos anteriores.
- Etiqueta tipo 4: este tipo es compatible con los estándares ISO/IEC 14443 Tipo A y Tipo B. Su memoria disponible es variable, llegando hasta 32 Kbyte por servicio.

En la tabla 2-1 se comparan las principales características de las etiquetas especificadas por el *NFC Forum*.

Parámetros	Tipo 1	Tipo 2	Tipo 3	Tipo 4
Basado en el estándar	ISO/IEC 14443 Tipo A	ISO/IEC 14443 Tipo A	FeliCa	ISO/IEC 14443 Tipo A, Tipo B
Nombre del chip	Topaz	MIFARE	FeliCa	DEsfire, SmartMX-JCOP
Tamaño de la memoria	Hasta 1 Kb	Hasta 2 Kb	Hasta 1 Mb	Hasta 64 Kb
Tasa de datos	106 kbits/s	106 kbits/s	212 kbits/s	106-424 kbits/s
Costo	Bajo	Bajo	Alto	Medio/Alto
Seguridad	Firma digital de 16 o 32 bits	Sin seguridad	Firma digital de 16 o 32 bits	Variable
Casos de uso	Etiquetas con memoria reducida para una sola aplicación		Etiquetas de mayor memoria para una sola aplicación	

Tabla 2-1. Comparación de los tipos de etiquetas NFC.

Fuente: Coskun V., Ok K., Ozdenizci B. (2013). *Professional: NFC Application Development for Android™*.

II.1.4.10 Formato de Intercambio de Datos NFC o *NFC Data Exchange Format* (NDEF)

NDEF es un formato de datos para intercambiar información entre dispositivos NFC, es decir, entre un dispositivo NFC activo y una etiqueta pasiva o entre 2 dispositivos NFC activos. Este formato es el mismo tanto para tarjetas como para dispositivos móviles, de esto se concluye que la información de NDEF no guarda relación con el tipo de dispositivo que participa en la comunicación. La especificación de NDEF es un estándar definido por el *NFC Forum*.

NDEF es un formato de mensaje binario que encapsula uno o más paquetes de carga útil definida por las aplicaciones en un solo mensaje. Un mensaje NDEF contiene uno o más registros NDEF. Cada registro consiste en una carga útil o *payload* de hasta $2^{32} - 1$ octetos de tamaño. Se pueden encadenar registros para soportar una carga útil más grande.

II.1.4.11 Definición de Tipo de Registro o Record Type Definition (RTD)

Esta definición proporciona las pautas para la especificación de los tipos de registros, que pueden ser incluidos en mensaje NDEF.

El *NFC Forum* define dos tipos: *NFC Forum External Types* y *NFC Forum Well-Known Types*, siendo el primero creado para dar a otros organismos la posibilidad de especificar sus propios tipos de forma independiente. ^[1]

Con respecto al *NFC Forum Well-Known Type*, es necesario decir que fue estandarizado por las especificaciones del *NFC Forum* y que proporciona la pauta para el procesamiento y representación de los datos. Ellos son: ^[1]

- *Text Record Type*: contiene solo texto simple, ninguna aplicación específica asignada.
- *Uniform Resource Identifier (URI) Record Type*: puede contener direcciones de correo electrónico, direcciones de Internet, números de teléfono u otros códigos de identificación.
- *Smart Poster Record Type*: es una extensión del tipo de registro URI, que proporciona información adicional acerca del URI, como iconos o acciones recomendadas.
- *Generic Control Record Type*: proporciona una estructura para cualquier actividad de control.
- *Signature Record Type*: contiene una firma prevista para certificar la veracidad de los datos.
- *Connection Handover*: ofrece traspaso de una conexión NFC a otra tecnología de comunicación con mayor rendimiento de datos (por ejemplo, *Bluetooth*).

II.1.4.12 Aplicaciones de la tecnología NFC

Los modos de operación de la tecnología NFC tienen diferentes características por lo que cada uno puede proveer diferentes casos de uso. A continuación se describen algunas de las más recientes aplicaciones de la tecnología NFC dependiendo de su modo de operación.

- Aplicaciones del modo Lector/Escritor: en este modo, numerosas oportunidades de casos de uso están disponibles. Los datos transmitidos pueden ser texto, una URL, la identificación de un producto o algún otro tipo de datos. Después de la operación de

transferencia, el teléfono móvil puede utilizar los datos para muchos propósitos de acuerdo con el diseño del caso de uso.

Las empresas o profesionales pueden diseñar proyectos de este modo debido a la flexibilidad de diferentes tipos de datos que se almacenan en la etiqueta NFC, así como la flexibilidad sobre cómo utilizar esa información. Por lo tanto, una amplia gama de aplicaciones en la salud, la educación, los servicios basados en la localización, servicios remotos, redes sociales y entretenimiento potencialmente pueden ser generadas utilizando el modo de lector/escritor. ^[1]

- Aplicaciones del modo *Peer-to-Peer* o Entre pares: como se mencionó anteriormente, el modo de *Peer-to-Peer* proporciona el intercambio de datos entre dos teléfonos móviles NFC. El fácil intercambio de datos entre dos dispositivos NFC con capacidades de computación de alto rendimiento ofrece la posibilidad para un intercambio seguro de datos privados, aprovechando la potencia de procesamiento de cada dispositivo.

Estos pueden transferir datos a unos pocos centímetros, por lo que el intercambio de datos privados e importantes puede ser una de las principales aplicaciones de este modo. Emparejar dispositivos *Bluetooth*, intercambiar tarjetas de visita u otros datos, tales como ficheros, imágenes y/o mensajes de texto, hacer nuevos amigos en las redes en línea (Twitter, Facebook, etc) y juegos son ejemplos de posibles implementaciones de este modo de operación.

- Aplicaciones del modo de Emulación de tarjetas: algunos ejemplos que ilustran cómo se puede implementar el modo de emulación de la tarjeta en diversas aplicaciones, son presentados a continuación:
 - Pago: se pueden tener diferentes tipos de aplicaciones de pago NFC. No hay duda de que las opciones de pago más importantes son las aplicaciones de tarjetas de crédito y débito, las cuales pueden ser activadas por los lectores NFC. Otras oportunidades para el pago vía NFC incluyen el almacenamiento y el uso de vales, tarjetas de regalo, etc.
 - Lealtad: los usuarios pueden ganar puntos de fidelidad en los puntos de pago y posteriormente pueden utilizarlos para hacer compras libres u obtener un

regalo. Además, los lectores NFC que utilizan este modo de funcionamiento se pueden usar para validar los cupones descargados a través de *Smart Posters*.

- Venta de entradas: los casos de uso de venta de entradas pueden ser implementadas en diferentes formas. Los usuarios pueden almacenar diferentes tipos de tickets como, como tickets de teatro, tickets de autobús, y los boletos aéreos descargados anteriormente a través de *Smart Poster* o por algún otro método. Los usuarios pueden utilizar estas entradas en torniquetes o puntos de venta a través del modo de emulación de la tarjeta.
- Control de acceso: en este caso los usuarios pueden insertar sus objetos de control de acceso en sus dispositivos móviles. Ejemplos de estos casos incluyen llaves electrónicas para coches, edificios, áreas seguras, y habitaciones de hotel.
- El registro en el hotel es un caso interesante ya que permite a los usuarios recibir una llave de la habitación a través de la tecnología OTA (*Over The Air*) antes de llegar al hotel y que permite a los usuarios registrarse directamente en la habitación. De esta manera, no hay ninguna razón para parar en la zona de recepción al llegar al hotel.
- Servicios de identificación: otro caso de uso interesante de este modo es el almacenamiento de información relacionada con la identificación dentro de los dispositivos móviles y permitiendo solo al personal autorizado acceder a ella. Un ejemplo de este tipo de servicio es el almacenamiento de los datos del paciente. La historia médica de un paciente se puede almacenar en un dispositivo móvil, y el usuario puede optar por dar permiso a un médico para tener acceso a esa información a través de un lector de NFC. Los datos médicos no necesitan ser almacenados en el hospital o en los servidores de la compañía de seguros médicos. Este caso de uso aumenta la privacidad del usuario, ya que terceros no deseados, como compañías de seguros no son capaces de acceder a la información del paciente. Se pueden desarrollar aplicaciones aún más interesantes en esta categoría, como la integración de las tarjetas de identificación nacional, pasaportes, huellas dactilares y licencias de conducir a los teléfonos móviles.

II.1.4.13 Ventajas y desventajas de la tecnología NFC

Debido a que el caso de uso con mayor desarrollo en la actualidad es el del pago, la mayoría de las ventajas y desventajas estarán enfocadas en este. Sin embargo, algunas pueden ser fácilmente extrapoladas a casos de uso con características similares como control de acceso e identificación, entre otros.

- Ventajas ^[1] ^[2]
 - Conveniente: la conveniencia de pago es una de las mayores ventajas de este sistema. NFC hace que sea muy fácil para los usuarios hacer el pago inmediato a través de sus teléfonos inteligentes y tabletas, con su billetera móvil. Este proceso de pago también es fácil de entender y usar. Ayuda a los usuarios a realizar transacciones financieras con el simple toque de su pantalla.
 - Versátil: NFC es muy versátil, ya que cubre una amplia gama de diferentes industrias y servicios. Esta forma de pago puede ser usado para los propósitos de la banca móvil, reservando plazas de restaurantes y pases para el cine, billetes de tren de reserva, obtener actualizaciones en tiempo real sobre el gasto y puntos de recompensa, canjear cupones y mucho, mucho más.
 - Mejora la experiencia del usuario: las empresas que adoptan fácilmente las nuevas tecnologías son consideradas por sus clientes como dinámicas y progresivas. Ofrecer una mejor experiencia de usuario ayuda a los establecimientos a mejorar su propia productividad y la eficiencia, lo que les permite mantener la fidelidad de los clientes, al tiempo que atrae a los clientes más nuevos.
 - Permite una comunicación fluida o sin fisuras: NFC se ha encontrado beneficioso en el ámbito académico. El alto nivel de encriptación permite a las instituciones de emplearlo como una especie de sistema de seguridad, lo que hace una identificación precisa de los estudiantes que entran y salen de las instalaciones. Los empleados de las empresas utilizan esta tecnología para interactuar a la perfección en el entorno de la oficina, el intercambio de información en tiempo real entre sí.

- Aporta seguridad cuando es utilizado como método de pago: el uso de billeteras móviles es, hasta cierto punto, más seguro que el uso de tarjetas de crédito físicas. En el desafortunado caso de robo del dispositivo móvil, la información de tarjeta de crédito del usuario y su contraseña, estarán protegida por PIN. Esto añade en una capa adicional de seguridad. Además, las tarjetas de pago con NFC se construyen para ser más seguras que las tradicionales de banda magnética. Otro aspecto de seguridad para el cliente es que durante el uso de este sistema de pago, los minoristas no tendrían acceso físico a la información de tarjetas de crédito de los clientes.
- Desventajas ^[1] ^[2]
 - Costosa: puede llegar a tener un costo muy elevado para las pequeñas empresas adoptar la tecnología NFC derivado de la compra y mantenimiento las máquinas y otros equipos relacionados. Mientras que las compañías grandes y bien establecidas, como Starbucks™ han incorporado con éxito la tecnología en sí misma; las empresas más pequeñas podrían tener dificultades para mantener su volumen de negocio existente y aumentar las ganancias. Este costo elevado para algunas pequeñas empresas ha limitado la expansión de esta tecnología en ese sector.
 - Posee un nivel de seguridad medio: si bien las transacciones NFC son, sin duda, más seguras que los pagos regulares con tarjetas de crédito, esta tecnología no está completamente libre de riesgos. La rápida evolución de la tecnología siempre viene con una igualmente poderosa consecuencia negativa. El *Hackeo* de teléfonos móviles es ahora rampante y los atacantes están saliendo con nuevos métodos para obtener acceso no autorizado a los datos personales de los usuarios, la seguridad social y datos financieros almacenados en el mismo. Esto hace que todo el sistema sea vulnerable e inseguro. La evidente falta de seguridad podría desalentar a los usuarios y empresas de calentamiento a esta tecnología en un futuro próximo.

II.2 Sistema Operativo Android

Android es actualmente una de las plataformas móviles más populares en el mundo, siendo utilizada también por teléfonos celulares. Internamente, es una versión modificada de Linux con una interfaz de programación integrada usando el lenguaje Java.

Android es propiedad de la *Open Handset Alliance OHA*, y es un proyecto completamente de código abierto. Actualmente, la empresa Google, una de las más importantes del panorama tecnológico mundial, lidera el desarrollo del proyecto, y la mayoría del código es liberado bajo la licencia Apache, la cual permite al usuario del software la libertad de usarlo para cualquier propósito, distribuirlo, modificarlo, y distribuir versiones modificadas de ese software. ^[22]

II.2.1 Características principales del sistema operativo Android

- Interfaz táctil : La interfaz de usuario predeterminada del sistema operativo Android está basada en la manipulación directa, usando entradas por toques, similares a acciones del mundo real como deslizar, toques, pellizcos y estiramientos para manipular los elementos en la pantalla y en un teclado virtual. La respuesta al usuario está diseñada para ser inmediata y proveer una interfaz táctil fluida, usando a menudo las capacidades vibratorias del dispositivo para producir lo que se conoce como *feedback* háptico al usuario, que básicamente es recrear la sensación de toque de elementos con volumen. Los componentes del hardware interno como acelerómetros, giroscopios y sensores de proximidad son usados por algunas aplicaciones para responder a acciones adicionales del usuario, como por ejemplo el ajuste de la pantalla de modo vertical a apaisado u horizontal dependiendo de la orientación que el usuario le dé al dispositivo.^[12]
- Aplicaciones: Android tienen una creciente selección de aplicaciones de terceros, que pueden ser adquiridas por los usuarios a través de una tienda de aplicaciones como la Google Play o la Amazon *Appstore*, o descargando e instalando el *Application Package File* (APK, por sus siglas en inglés) desde algún sitio web.

Las aplicaciones comúnmente llamadas “*apps*”, abreviación de *Application* (Aplicación en inglés) que extienden las funcionalidades del dispositivo, son

desarrolladas principalmente en el lenguaje de programación java usando el Kit de Desarrollo de Software de Android o Android Software Development Kit (SDK).

- Manejo de memoria: Dado que el sistema operativo Android es utilizado en dispositivos alimentados por baterías, está diseñado para administrar la memoria principal de los dispositivos de forma que logre mantener un consumo mínimo de energía, en contraste con sistemas operativos de escritorio los cuales generalmente asumen que están conectado a una fuente ilimitada de electricidad. Cuando una aplicación de Android deja de ser utilizada, el sistema automáticamente la suspenderá en memoria, es decir, aunque la aplicación sigue técnicamente “abierta”, en el modo suspendido esta no consume recurso y se queda esperando en segundo plano hasta que sea necesitada de nuevo.

Por ser un sistema diseñado para funcionar en una amplia gama de dispositivos móviles también cuenta con las siguientes características:

- Conectividad: Soporta conexiones utilizando las tecnologías GSM/EDGE, Wi-Fi, *Bluetooth*, LTE, CDMA, EV-DO, UMTS, NFC, IDEN and WiMAX. Además permite utilizar el dispositivo como un punto de acceso a redes inalámbricas.
- Soporte multimedia: Soporta los principales formatos de audio y video, entre los que destacan: WebM, H.263, H.264, AAC, MPEG-4 SP, MP3, MIDI, Ogg Vorbis, FLAC, WAV, JPEG, PNG, GIF, BMP y WebP.
- Almacenamiento externo: Permite el uso de tarjetas de memoria para almacenar contenido.

II.2.2 Componentes principales del sistema operativo Android

- *Kernel* de Linux: actúa como una capa de abstracción entre el hardware y la pila de software. Android contienen un *kernel* de Linux versión 2.6 para proveer servicios del núcleo del sistema. A partir de la versión 4.0 de Android, la versión utilizada paso a ser la 3. ^[1]
- Tiempo de ejecución de Android: Android incluye un conjunto de bibliotecas esenciales que tienen la mayoría de las funcionalidades disponibles en las bibliotecas base del

lenguaje Java. Cada aplicación Android ejecuta su propio proceso en su instancia de la Máquina Virtual Dalvik o Dalvik Virtual Machine DVM. ^[1]

- **Bibliotecas:** Android incluye un conjunto de bibliotecas escritas en los lenguajes C y C++ que proveen de funcionalidades al sistema Android. Estas habilidades o características son otorgadas a los desarrollares a través del *framework* de aplicaciones. ^[1]
- **Framework de aplicaciones:** debido a que Android provee una plataforma de desarrollo abierta, ésta permite a los desarrolladores construir aplicaciones innovadoras y dinámicas. Los desarrolladores pueden beneficiarse de los diferentes componentes y servicios de Android, como acceder al hardware del dispositivo, usando información de ubicación, ejecutando servicios en segundo plano, configurando alarmas y agregando notificaciones a la barra de estado, entre otras posibles acciones. ^[1]
- **Aplicaciones:** Android incluye un conjunto de aplicaciones esenciales de fábrica tales como un cliente de correo electrónico, un programa SMS, calendario, mapas, navegador y agenda de contactos. ^[1]

II.2.3 Dispositivos móviles con sistema operativo Android equipados con tecnología NFC

La lista de teléfonos inteligentes o *smartphones* y tabletas o *tablets* que incluyen características de la tecnología NFC es realmente extensa. A continuación se mencionan algunos de los modelos más destacados de las principales empresas del sector: ^[19]

- Motorola moto G XT1058, moto X XT1058, Droid Ultra X1080, Razr i XT8903
- LG Optimus. Modelos: L5 E612/E615, L7 P700/P705, F6 D500, L9 P760/778g,G E970/E973, 4X HD P8805
- LG G2 y G2 MINI
- Samsung Galaxy. Modelos: Ace 2, Fame, Mini II (no todas las versiones), Nexus, S II (no todas las versiones), S III, S III Mini (no todas las versiones), S IV, S IV (Mini), S V , S V Mini (no todas las versiones), Note (no todas las versiones), Note II, Note III
- Google Nexus. Modelos: S, 4, 5, 7, 10
- HTC Amaze 4G

- HTC One y One X
- Huawei Ascend. Modelos: G615 (HONOR 2) , G630, G300 (no todas las versiones), G510 (no todas las versiones)
- Sony Xperia. Modelos: S, Acro S, sola, L, M, P, ZL, SP, Z, ion, GX, T, V, ZR, Z1, M2, Z2.
- Panasonic Eluga
- XIAOMI MI3, MI2A

En los dispositivos de la lista anterior, basta con activar la conectividad NFC a través del menú de configuraciones de Android e instalar una aplicación que permita hacer uso de las funcionalidades que incorpora el dispositivo.

II.3 Aplicaciones Móviles

En esta sección se definen conceptos asociados con las aplicaciones móviles, su clasificación y los componentes necesarios para su desarrollo en la plataforma Android.

II.3.1 Definición de aplicación móvil

Una aplicación móvil, más comúnmente conocida como app, es un tipo de software diseñado para ejecutarse en un dispositivo móvil como un teléfono inteligente o *smartphone*, o en una computadora tipo *tablet*. Las aplicaciones móviles son usadas frecuentemente para ofrecer a los usuarios servicios similares a los que se pudiesen acceder desde un computador de escritorio. Las apps son generalmente pequeñas unidades de software individual con funciones limitadas. ^[13]

II.3.2 Arquitectura de las aplicaciones móviles

Decidir la arquitectura de la información es fundamental cuando se desea desarrollar una aplicación móvil ^[14], pudiendo esta ser auto contenida (sin conexión) o en cambio si el contenido se obtiene de alguna fuente externa.

- Aplicación móvil auto contenida: En este caso toda la información utilizada y mostrada por la aplicación esta almacenada dentro de los archivos de esta y no es necesario descargar algún contenido extra.

Estas aplicaciones son fáciles de desarrollar y no hay costes añadidos por contrataciones de almacenamiento en línea. Sin embargo, si en algún momento es necesario modificar la información contenida en la aplicación deberá compilarse una nueva versión con la información actualizada y distribuirla nuevamente, proceso que pudiese llegar a ser inviable si el volumen de información es muy alto.

- Aplicación móvil con conexión u *on-line*: Para este tipo de aplicaciones la información se encuentra alojada en una base de datos o en fichero XML (*eXtensible Markup Language*) alojado en algún servidor web remoto de forma que la debe realizarse una conexión al iniciarse la aplicación y descargar la información necesaria.

II.3.3 Clasificación de las aplicaciones móviles

Las aplicaciones que se ejecutan en los dispositivos móviles pueden clasificarse como nativas, web o híbridas. ^[13]

- Aplicación Nativa: Está programada en un lenguaje específico y utilizando APIs propias de la plataforma. Se suele comprar, descargar y actualizar a través de la tienda de aplicaciones específica de la plataforma. Las aplicaciones nativas suelen ofrecer mejor rendimiento, integración más completa y la mejor experiencia de usuario en comparación con otras opciones; sin embargo, el desarrollo nativo suele ser también la opción de desarrollo más compleja.
- Aplicación Web: Una aplicación web (*web-app/web-based*) se basa en tecnologías web como HTML5 (*HyperText Markup Language versión 5*), JavaScript y CSS (*Cascade Style Sheet*), y no depende de ninguna tienda de aplicaciones. Es un sitio almacenado localmente en el móvil que trata de emular el *look-and-feel* (aspecto y comportamiento) de una aplicación. Hay *frameworks* para aplicaciones web que permiten construir un contenedor nativo para tales aplicaciones de manera que puedan ser publicadas en las tiendas de aplicaciones, por ejemplo Phonegap.

- Aplicaciones Híbridas: Existe una exagerada controversia en torno a si las aplicaciones nativas son el futuro o si lo son las aplicaciones web. Para muchos de los desarrolladores de aplicaciones móviles esta controversia ya no es relevante, porque la aproximación híbrida al desarrollo de aplicaciones se ha convertido en algo bastante común: una aplicación puede utilizar código nativo para incrementar su rendimiento e integrarse con la plataforma, mientras que utiliza una *webview* o vista web con contenido basado en HTML5 para otras partes de dicha aplicación.

Una aplicación híbrida hace uso tanto de las tecnologías nativas como las web. Partes de ella se comportarán como una aplicación nativa, mientras que otras se ejecutan sobre tecnologías web y pueden utilizar la conexión a Internet para ofrecer contenidos actualizados y funcionalidades sin necesitar de publicar actualizaciones en las tiendas de aplicaciones. Mientras que esto podría ser visto como un inconveniente, el uso de las tecnologías web permite a los desarrolladores mostrar contenido actualizado sin tener que volver a subir la aplicación a las tiendas de aplicaciones. El desafío clave es combinar las capacidades únicas de tecnologías nativas y web para crear una aplicación verdaderamente fácil de usar y atractiva.

II.3.4 Desarrollo de aplicaciones móviles para el sistema operativo Android

Actualmente Google ofrece a los interesados en desarrollar aplicaciones para el sistema operativo Android el Kit de desarrollo de software de Android o *Android Software Development Kit* (SDK, por sus siglas en inglés) el cual provee un extenso conjunto de herramientas incluyendo un depurador, bibliotecas, emuladores de dispositivos, documentación, código de ejemplo, y tutoriales. Las aplicaciones pueden ser fácilmente desarrolladas utilizando la plataforma oficial de desarrollo, que incluye los IDE (*Integrated Development Enviroment* o Ambiente de desarrollo integrado) Eclipse y Android Studio, el cual está basado en IntelliJ; ambos en conjunto del *plug-in* ADT (*Android Development Tools*). Utilizar estas herramientas ahorra algo de tiempo al momento de mantener actualizadas las versiones del SDK y ofrece una experiencia más orientada a Android. ^[13]

Para desarrollar aplicaciones para Android es necesario instalar el Kit de Desarrollo de Java o *Java Development Kit* (JDK) y el ADT. Previa a la instalación de estas herramientas es necesario instalar el Ambiente de Ejecución de Java o *Java Runtime Enviroment* (JRE).

Estos componentes del lenguaje de programación Java son necesarios debido a que las aplicaciones que se desean desarrollar deben ser escritas en este lenguaje. Java además de un lenguaje es una tecnología multiplataforma de programación orientada a objetos. Fue presentada en 1995 por Sun Microsystems.

II.3.4.1 Implementación de una aplicación

- Arquitectura de la aplicación: las aplicaciones Android habitualmente incluyen una mezcla de múltiples tipos de objetos como actividades (*Activity*), servicios (*Service*), Receptores de avisos (*BroadcastReceiver*) y proveedores de datos (*Data Provider*), los cuales tienen que ser declarados en el manifiesto de aplicación o *Android Manifest*, un fichero XML con la información esencial de la aplicación. Una actividad es un bloque de funcionalidad con una interfaz de usuario adjunta. Un servicio se utiliza para tareas que se ejecutan en segundo plano y, por lo tanto, éstas no están ligadas directamente a una representación visual. Un receptor de mensajes gestiona los mensajes transmitidos por el sistema, por el usuario o por otras aplicaciones. Un proveedor de datos es una interfaz para el contenido de una aplicación que se abstrae de los mecanismos de almacenamiento subyacentes (por ejemplo, SQLite).

Una aplicación puede consistir en varios de estos componentes, por ejemplo, una actividad para la interfaz de usuario y un servicio para tareas dilatadas en ejecución. La comunicación entre los componentes se consigue con objetos tipo Intención (*Intent*) o llamadas a procedimientos remotos vía el lenguaje de definición de interfaces de Android o *Android Interface Definition Language* (AIDL, por sus siglas en inglés). Un *Intent* empaqueta datos, tales como la localización del usuario o una URL, con una acción.

Estos *Intents* desencadenan acciones en la plataforma y pueden ser utilizados como un sistema de mensajería en tu aplicación. Por ejemplo, el *Intent* para mostrar una página web abrirá el navegador web. La fuerza de esta filosofía de bloques de construcción es que se puede sustituir cualquier funcionalidad por otra aplicación, ya que Android siempre utiliza la aplicación por defecto para un propósito específico. Por ejemplo, el *Intent* para compartir una página web llamada por una aplicación de lectura de noticias puede abrir un cliente de correo electrónico o una aplicación de mensajería

de texto, dependiendo de las aplicaciones instaladas y de las preferencias de usuario: se puede utilizar cualquier aplicación que declare un *Intent* tipo share como interfaz.

La interfaz de usuario de una aplicación está separada del código mediante archivos XML de *layout* específicos de Android. Diferentes diseños pueden ser creados para diferentes tamaños de pantalla, localización por país y características del dispositivo, sin tocar el código Java. Con este fin, textos e imágenes están organizados en carpetas separadas de recursos. Por supuesto, también se puede definir y diseñar *layouts* vía código o hacer uso de ambas estrategias para permitir actualizaciones dinámicas de la interfaz de usuario.

- Pruebas: el primer paso para probar una aplicación es ejecutarla en el emulador provisto por el ADT o en un dispositivo. Es posible depurarla, si es necesario, a través de la herramienta *Dalvik Debug Monitor Service* o DDMS. Todas las versiones del sistema operativo Android están preparadas para ejecutarse en dispositivos sin necesidad de modificaciones aunque, sin embargo, algunos fabricantes de *hardware* pueden realizar cambios en algunos elementos de la plataforma; por tanto, probar la aplicación en varios dispositivos es esencial.

II.3.5 API de Android para NFC

Los dispositivos móviles que incluyen el sistema operativo Android y tecnología NFC integrada soportan los tres modos de operación mencionados anteriormente: Modo Lector/escritor, Modo *Peer-to-Peer* y Modo de Emulación de tarjetas Inteligentes. ^[15]

Android básicamente puede trabajar con datos NDEF, o formato de mensaje binario que encapsula uno o más paquetes en un solo mensaje, de dos formas:

- Leyendo datos NDEF de una etiqueta NFC
- Emitir mensajes NDEF desde un dispositivo con Android Beam™ a otro.

Además ofrece una API que permite a los desarrolladores tener acceso a los componentes de la tecnología NFC incorporados en los dispositivos móviles, tanto a nivel de hardware como de software. Dicha interfaz de programación de aplicaciones incluye un conjunto de funciones y recursos que facilita el trabajo del programador.

A continuación se describen los métodos implementados por el sistema operativo Android para establecer la comunicación mediante tecnología NFC entre dispositivos compatibles:

II.3.5.1 Leyendo datos NDEF de una etiqueta NFC

La lectura de los datos NDEF es manejada por el Sistema de Despacho de Etiquetas (*Tag Dispatch System*) el cual analiza las etiquetas NFC descubiertas, categoriza los datos apropiadamente e inicia una aplicación que esté interesada en los datos previamente categorizados.

Una aplicación que quiera manejar la etiqueta NFC escaneada puede declarar un filtro de intento (intent filter) y una solicitud para manejar los datos.

La característica de Android Beam™ permite a los dispositivos compartir mensajes NDEF a través del contacto físico entre estos. Esta interacción provee una forma más sencilla de enviar datos, si se compara con otras tecnologías inalámbricas como el *Bluetooth*, ya que con NFC no es necesario realizar un emparejamiento manual de los dispositivos.

II.3.5.2 El Sistema de Despacho de Etiquetas

Los dispositivos con Android suelen buscar etiquetas NFC cuando la pantalla está desbloqueada, a menos que la conexión NFC está desactivada en el menú Ajustes del dispositivo. Cuando un dispositivo con Android descubre una etiqueta NFC, el comportamiento deseado es tener la actividad más apropiada manejar la intención sin preguntar al usuario qué aplicación utilizar. Dado que los dispositivos escanean las etiquetas NFC en un rango muy corto, si los usuarios tuvieran que seleccionar manualmente una actividad, estarían obligados a mover el dispositivo lejos de la etiqueta y romper la conexión. Se debe desarrollar una actividad para manejar sólo las etiquetas NFC de interés y así prevenir que aparezca el Selector de actividades.

Android ofrece un sistema de despacho de etiqueta especial que analiza las etiquetas NFC escaneadas, luego los analiza gramaticalmente, y trata de localizar las aplicaciones que estén interesados en los datos escaneados. A grandes rasgos, lo hace de la siguiente forma:

1. Analiza la etiqueta NFC y averigua el tipo MIME o el URI (*Universal Resource Identifier*) que identifica la carga de datos en la etiqueta.

2. Encapsula el tipo MIME o URI y la carga útil en una intención o *Intent*.
3. Inicia una actividad basada en la intención.

II.4 Antecedentes de aplicaciones para Android que utilizan características NFC

A continuación se describirán casos exitosos de aplicaciones para el sistema operativo Android que utilizan las funcionalidades NFC incorporadas. Estos servirán como referencia del estado actual del desarrollo en esta área además de aportar bases para estructuración y diseño de la aplicación que se desea desarrollar.

- Aplicación Trigger: ésta aplicación interactúa con el entorno para configurar automáticamente ajustes del teléfono. Combina funciones de activación y acciones para crear tareas y, a continuación, activa las tareas a partir de las condiciones indicadas por el usuario. [16]

Algunos ejemplos de lo que podrás hacer:

- En el coche: usar la conexión *Bluetooth* para abrir el GPS y activar una aplicación de música.
- En la mesita de noche: programar una etiqueta NFC para que la alarma vibre, se apague la pantalla y ajuste una alarma.
- En casa: configurar la desactivación de los datos móviles cuando el teléfono detecte la red Wifi.

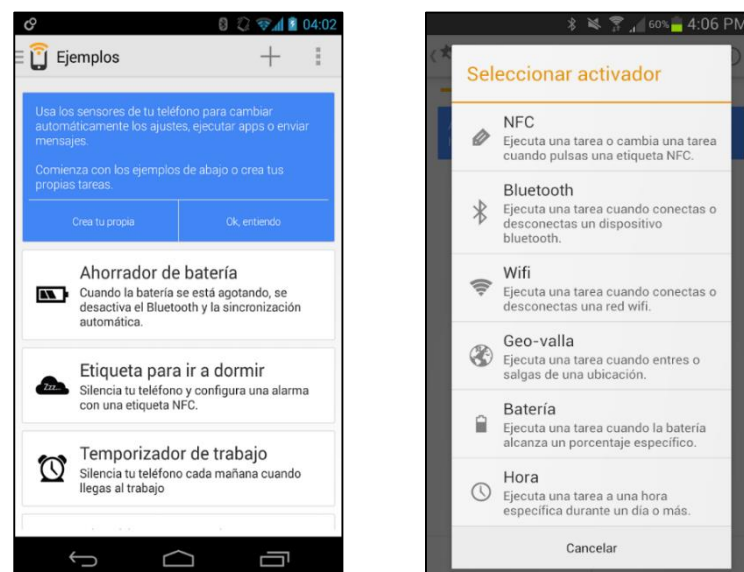


Figura 2-3. Interfaz de usuario de la aplicación NFC Trigger.
Fuente: Google Play Store, Aplicación NFC Trigger.

- Aplicación NFC Tools: ésta aplicación permite leer y escribir las etiquetas NFC y otros chips RFID compatibles con un dispositivo con tecnología NFC integrada. Al acercar una etiqueta NFC al dispositivo la aplicación le mostrará las informaciones disponibles, tal como: el fabricante de la etiqueta, el tipo de etiqueta, la norma de la etiqueta, la tecnología disponible, su número de serie, el tamaño de la etiqueta y de los datos presentes, si la escritura es posible y si la etiqueta permite la configuración en sólo lectura así como todos los registros presentes en la etiqueta al formato NDEF.

También puede escribir diferentes tipos de datos en sus etiquetas NFC: Texto, enlace web, enlace hacia una aplicación, un e-mail, un contacto, un número de teléfono, un SMS, una ubicación GPS, una dirección, parámetros de una conexión *Bluetooth*, datos personalizados, entre otros. La función de escritura permite añadir tantos datos como desee. Así, es posible registrar una gran cantidad de informaciones sobre su etiqueta. [17]

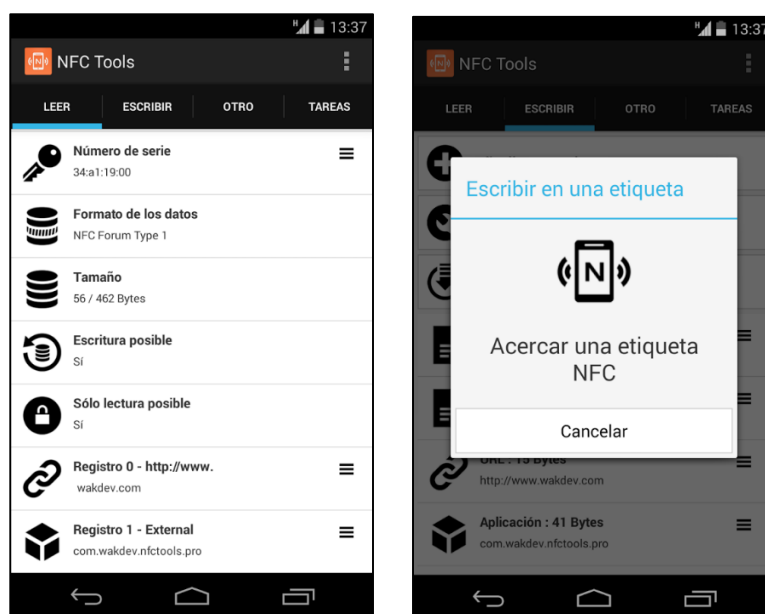


Figura 2-4. Interfaz de usuario de la aplicación NFC Tools.
Fuente: Google Play Store, Aplicación NFC Tools.

- Aplicación NFC TagWriter by NXP: ésta aplicación almacena los contactos, marcadores, ubicaciones geográficas, parámetros de conexiones vía *Bluetooth*, SMS, correos, mensajes de texto y mucho más en cualquier etiqueta NFC, así como a elementos como carteles, tarjetas de visita, relojes y muchos productos electrónicos que posean

tecnología NFC. Una vez que los datos se han almacenado la aplicación permite también la lectura y la visualización de los datos programados incluyendo opciones para iniciar aplicaciones automáticamente en función de los datos contenidos. [18]

- Características principales:
 - Orientación gráfica de usuario intuitiva para todas las funciones soportadas
 - Emparejamiento de *Bluetooth* para todas las versiones de dispositivos Android con NFC
 - Aplicación de lanzar con la opción de transferencia de datos
 - Crea fácilmente contenido de la etiqueta de sus contactos y favoritos
 - Crear un nuevo contenido con el editor de conjunto de datos NFC incluido
 - Convertir los códigos QR en conjuntos de datos NFC
 - Ver el contenido existente de una etiqueta
 - Exportación, importación y compartir conjuntos de datos NFC
 - Gestionar su propia historia de las etiquetas que ha visto y escrito
 - Función para ejecutar el contenido de las etiquetas NFC con un simple toque
 - Copia de seguridad de los contenidos de una etiqueta antes de escribirlo
 - Borrar el contenido de una etiqueta
 - Protección contra escritura una etiqueta

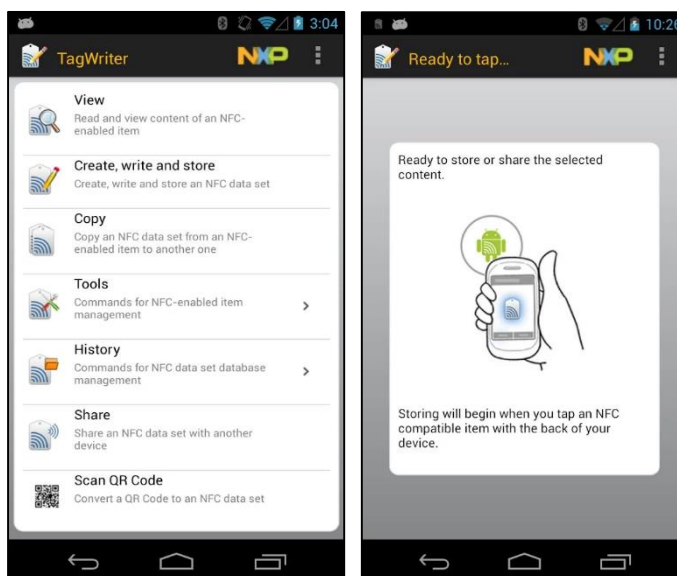


Figura 2-5. Interfaz de usuario de la aplicación NFC TagWriter by NXP.
Fuente: Google Play Store, Aplicación NFC TagWriter by NXP.

II.5 Conclusiones de la investigación

En base a la investigación realizada, es posible determinar que el enfoque de desarrollo que mejor se ajusta a los requerimientos de ALS TELECOM implica lo siguiente:

- Considerar el Sistema Operativo Android como plataforma de despliegue, debido a su amplia presencia en el mercado de dispositivos móviles, ubicándose por delante de plataformas como Windows Phone que también ofrecen dispositivos compatibles con la tecnología NFC.
- La implementación del modo de operación Lector / Escritor estipulado por la tecnología NFC, en lugar de los modos Emulación de Tarjeta Inteligente y *Peer-to-Peer* (P2P).
- El desarrollo de una aplicación nativa, con el fin de maximizar el rendimiento de las funcionalidades NFC y multimedia garantizadas por las herramientas de programación provistas por la plataforma.
- Una aplicación orientada a conexión, que si bien esta ideada para recibir contenido desde etiquetas NFC también ofrecerá la posibilidad de generar y almacenar contenido.

CAPÍTULO III: MARCO METODOLÓGICO

El desarrollo de la aplicación planteada para resolver el problema de investigación se llevó a cabo siguiendo una metodología ad-hoc, basada en el Desarrollo Incremental e Iterativo.

III.1 Desarrollo Incremental e Iterativo

El modelo de Desarrollo Incremental fue propuesto por Harlan Mills en el año 1980, como una forma de reducir la repetición del trabajo en el proceso de desarrollo y dar oportunidad de retrasar la toma de decisiones en los requisitos hasta adquirir experiencia con el sistema. Durante el proceso se trata de llevar a cabo el proyecto en diferentes partes o módulos que al final terminará siendo la solución completa requerida por el cliente. Pero éstas partes no se pueden realizar en cualquier orden, sino que dependen de lo que el cliente esté necesitando con más urgencia, de los puntos más importantes del proyecto, los requerimientos más básicos, difíciles y con mayor grado de riesgo, ya que estos se deben desarrollar al comienzo de manera que se disminuya la dificultad y el riesgo en cada versión.

[23]

De este modo se puede conseguir una aplicación ejecutable (primera versión) que podrá ser entregada al cliente para que éste pueda trabajar en ella y el programador pueda considerar las recomendaciones que el cliente efectúe para hacer mejoras en el producto. Estas nuevas mejoras deberán esperar a ser integradas en la siguiente versión junto con los demás requerimientos que no fueron tomados en cuenta en la versión anterior.

El Desarrollo Iterativo o Evolutivo, por otra parte, es una derivación del modelo en cascada puro, el cual es un enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, siendo necesario terminar una etapa antes de comenzar con la otra y sin posibilidad de regresar a una etapa anterior. Con el enfoque iterativo se busca reducir el riesgo que surge entre las necesidades del usuario y el producto final por malos entendidos durante la etapa de solicitud de requerimientos.

En cada iteración o repetición del desarrollo iterativo se reproduce el ciclo de vida del desarrollo en cascada a menor escala. Los objetivos de una iteración se establecen en función

de la evaluación de las precedentes. Desde el principio, al final de cada iteración se le entrega al cliente una versión completa y mejorada del producto. El cliente es quien luego de cada iteración evalúa el producto y lo corrige o propone mejoras. Estas iteraciones irán refinando el sistema y se repetirán hasta obtener un producto que satisfaga al cliente.

En un desarrollo incremental e iterativo el proyecto se planifica en diversos bloques temporales de duración variable, en los cuales se evoluciona el producto a partir de los resultados completados de los bloques previos, añadiendo nuevos requerimientos y/o mejorando los que ya fueron completados, enfatizando siempre los requerimientos en función al valor que aportan al cliente.

A nivel básico, el modelo sigue el ciclo de vida del desarrollo incremental, observado en la figura 3-1, que a su vez está basado en el modelo cascada, incluyendo en cada sub-fase o etapa la posibilidad de revisar o mejorar funcionalidades desarrolladas previamente. El desarrollo incremental e iterativo no está orientado a un tipo específico de proyecto, sino a cierto tipo de cliente, siendo realmente útil cuando el mismo necesita entregas rápidas, aunque sean parciales.

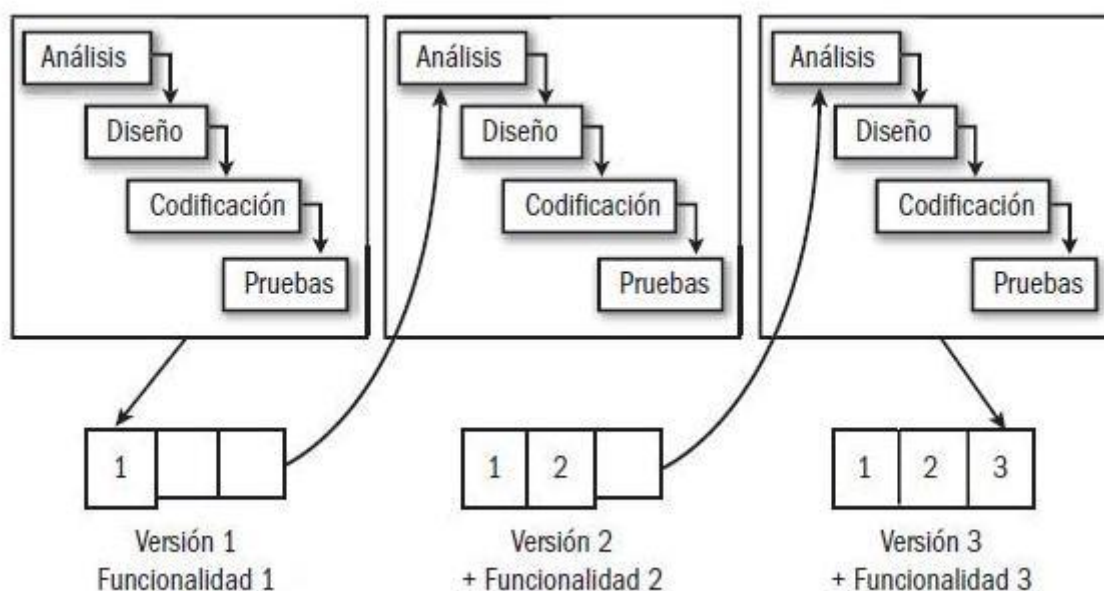


Figura 3-1. Ciclo de vida del desarrollo incremental. ^[20]
Fuente: Cantone, D. (2006). Implementación y Debugging.

III.1.1 Etapas del ciclo de vida del Desarrollo Incremental e Iterativo

Las fases o etapas que ocurren en el ciclo de vida del Desarrollo Incremental e Iterativo son enumeradas y descritas a continuación:

1. Etapa de inicialización

La meta de esta etapa es crear un producto con el que el usuario pueda interactuar, y por ende retroalimentar el proceso. Debe ofrecer una muestra de los aspectos claves del problema y proveer una solución lo suficientemente simple para ser comprendida e implementada fácilmente.

Para guiar el proceso de iteración se crea una lista de control de proyecto, que contiene un historial de todas las tareas que necesitan ser realizadas. Incluye nuevas funcionalidades para ser implementadas, y áreas de rediseño de la solución ya existente. Esta lista de control se revisa periódicamente como resultado de la fase de análisis.

2. Etapa de iteración

Esta etapa involucra el rediseño e implementación de una tarea de la lista de control de proyecto, y el análisis de la versión más reciente del sistema. La meta del diseño e implementación de cualquier iteración es ser simple, directa y modular, para poder soportar el rediseño de la etapa o como una tarea añadida a la lista de control de proyecto. En cada iteración se lleva a cabo el ciclo de vida del modelo de desarrollo Cascada el cual incluye las siguientes etapas:

- a. Análisis: esta fase involucra la elección de las tareas de la lista de control del proyecto que se desarrollaran en la presente iteración y la inclusión de nuevas actividades y mejoras a dicha lista.
- b. Diseño: esta fase envuelve entender los requerimientos del sistema y crear una idea general del producto final, definiendo adicionalmente la arquitectura del sistema.

- c. Codificación o Implementación: se desarrollan los diferentes módulos de la aplicación, de acuerdo a los requerimientos y la arquitectura descrita en las fases previas.
- d. Pruebas: se prueba la funcionalidad de cada módulo desarrollado, verificando que cada módulo cumpla las especificaciones establecidas.

III.1.2 Beneficios del Desarrollo Incremental e Iterativo

Entre las ventajas del Desarrollo Incremental e Iterativo se observan: ^[21]

- Se puede gestionar las expectativas del cliente de manera regular, pudiendo tomar decisiones en cada iteración, lo que resulta muy útil cuando el cliente no sabe exactamente que necesita o si se presenta la necesidad de realizar cambios a corto plazo.
- El cliente puede comenzar el proyecto con los requisitos de alto nivel que aportan mayor valor, incluso con requerimientos incompletos que pueden ser refinados en iteraciones sucesivas.
- El cliente puede obtener resultados importantes y usables desde las primeras iteraciones.
- Se puede gestionar de manera natural los cambios que van apareciendo durante el proyecto. La finalización de cada iteración es el lugar natural donde el cliente puede proporcionar sus observaciones tras examinar la versión producida y se pueden realizar cambios en la siguiente observación de acuerdo a las necesidades.
- Se minimiza el número de errores que se producen en el desarrollo y se aumenta la calidad del código, ya que cada iteración debe producir requisitos funcionales.

III.1.3 Restricciones del Desarrollo Incremental e Iterativo

Entre las desventajas que se pueden observar del Desarrollo Incremental e Iterativo se encuentran: ^[21]

- La disponibilidad del cliente debe ser alta durante todo el proyecto, dada su participación continua tanto al inicio como al final de cada iteración.

- Cada iteración debe dar como resultado requisitos terminados, de tal forma que, de encontrar dificultades, puede involucrar retrasos en las entregas o la necesidad de tiempo extra de desarrollo.
- La relación con el cliente debe estar basada en los principios de colaboración más que tratarse de una relación contractual en la cual cada parte defiende únicamente su beneficio a corto plazo.
- Es necesario disponer de técnicas y herramientas que permitan hacer cambios fácilmente en el producto, de manera que pueda crecer en cada iteración de forma incremental sin realizar un gran esfuerzo adicional.
- Aumenta la dificultad de sistemas de alto tamaño en los cuales se prevea realizar múltiples integraciones entre las partes, ya que se diseñan de forma separada.

III.1.4 Artefactos del Desarrollo Incremental e Iterativo

En base al Desarrollo Incremental e Iterativo se han creado varios *frameworks*, siendo el *Rational Unified Process* (RUP) uno de los más conocidos, que incluyen listas detalladas de artefactos a utilizar en cada fase del desarrollo. Sin embargo, la metodología indica únicamente el uso de la lista de control de proyecto, mencionada anteriormente, para guiar el proceso de las iteraciones.

Dada la importancia en el *feedback* del usuario en el Desarrollo Incremental e Iterativo, es habitual utilizar algún artefacto que recoja formalmente los resultados de las interacciones del mismo con cada entrega realizada al final de una iteración. Adicionalmente, se recomienda el uso de cualquier artefacto de diseño, codificación o pruebas para apoyar la ejecución exitosa de dichas fases, utilizando en cada iteración aquellos artefactos que sean de utilidad y descartando los que no sean útiles, así se hayan usado en una o más iteraciones previas.

III.1.5 Justificación

La aplicación que se desea desarrollar se basa en un conjunto de requerimientos altamente variables y en algunos casos vagamente definidos, por lo que según lo expuesto anteriormente en este capítulo, el Desarrollo Incremental e Iterativo supone una ayuda significativa para la implementación exitosa de la solución planteada.

La flexibilidad ante las iteraciones y los artefactos permite lidiar con los requerimientos de forma que sea de mayor utilidad para el cliente. Además, la frecuente generación de

prototipos funcionales consigue producir *feedback* de forma regular, por parte del cliente y algunos usuarios, para incorporar de forma eficiente las observaciones de estos.

III.2 Metodología a seguir para el desarrollo de la solución

La metodología a utilizar en el desarrollo de la aplicación sigue los principios del Desarrollo Incremental e Iterativo, bajo las siguientes consideraciones:

- Al iniciar el proceso de desarrollo se elaborará la Lista de Requerimientos o Lista de Control General que permitan obtener un prototipo funcional en la primera iteración. Al inicio de las posteriores iteraciones se escogerá un subconjunto de requerimientos de la Lista antes mencionada.
- Cada iteración, que representa un ciclo de vida del modelo Cascada, se plantea una duración que puede variar entre 1 y 5 días. En este periodo se realiza un análisis del requerimiento a implementar, se diseña, se codifica el requerimiento y se realizan pruebas de integración. En caso de ser necesario, se realizaran cambios al código existente para garantizar el perfecto funcionamiento del módulo recién agregado o para mejorar la calidad del producto general.
- Al finalizar cada iteración se revisará el prototipo funcional obtenido y se considerarán los nuevos requerimientos a incluir en la Lista de Control General así como la actualización del estado de los requerimientos implementados en la iteración más reciente.

II.2.1 Artefactos

Como parte de la documentación de los principales productos de las distintas etapas de la metodología, se utilizan los siguientes artefactos:

- Tablero de requerimientos: documento en formato de tabla que incluye los requerimientos funcionales de la aplicación. El tablero consta de cuatro columnas, como se muestra en la figura 4-1, que permiten observar el estado de desarrollo de una funcionalidad. Para que tanto el cliente como el desarrollador puedan comprobar el estado de desarrollo de la aplicación en cualquier momento y editarlo cuando se considere necesario se eligió la herramienta online Trello para realizar el tablero de requerimientos.

- Diagrama de secuencia: documento que refleja el flujo de comunicación entre los módulos de la aplicación.
- Diagramas de casos de uso: representación gráfica de la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas.
- Diagrama entidad-relación: representación gráfica de entidades y relaciones de los ficheros intercambiados a través de la aplicación.
- Diagrama del modelo Relacional: representación gráfica de los objetos de la base de datos.
- Lista de tareas: enumeración de las tareas a ser desarrolladas en la iteración actual y de las tareas propuestas para incrementos posteriores. Esta lista será actualizada conforme a los resultados obtenidos en cada fase de análisis o diseño. Se mantuvo la lista en un pizarrón situado en un lugar de fácil acceso, para poder actualizarla o agregarle elementos rápidamente.

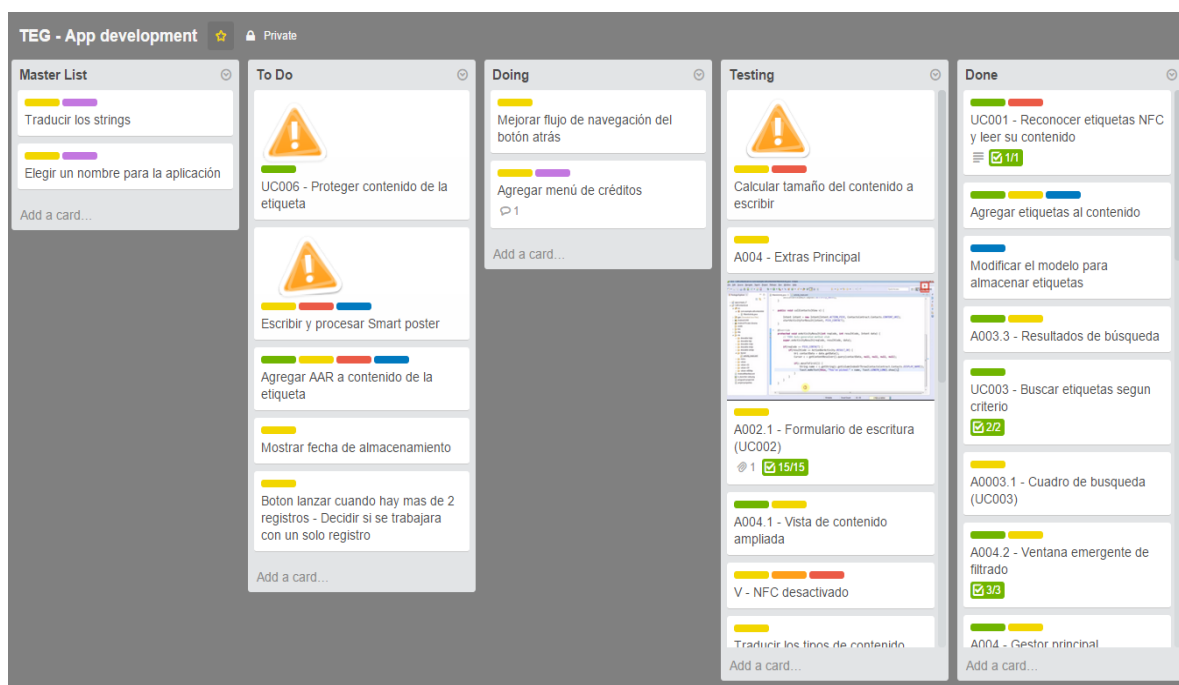


Figura 3-1. Tablero de requerimientos realizado en la herramienta online Trello.
Fuente: Creada por el autor.

II.2.2 Justificación

La metodología propuesta permite realizar entregas periódicas que agilizan la supervisión del estado de desarrollo del proyecto logrando desarrollar inicialmente los principales componentes de interacción entre el usuario y la aplicación, pudiendo así corregir rápidamente cualquier error grave de diseño o funcionamiento.

CAPÍTULO IV: Desarrollo de la Aplicación

Este capítulo ilustra el entorno de la aplicación, especificando sus objetivos generales y específicos, su alcance y la plataforma utilizada para su desarrollo e implementación. También presenta una descripción de la misma a mayor detalle, especificando los usuarios, los requerimientos, las principales funcionalidades aportadas, las interfaces principales, los casos de uso y las pruebas realizadas.

IV.1 Objetivo General de la aplicación

Permitir al usuario consumir, generar y compartir contenido de diversa índole entre dispositivos móviles con tecnología NFC integrada y etiquetas NFC.

IV.2 Objetivos Específicos de la aplicación

1. Presentar al usuario una interfaz atractiva que le permita consumir el contenido almacenado en una etiqueta NFC.
2. Ofrecer la posibilidad de generar nuevo contenido para ser almacenado en etiquetas NFC.
3. Permitir al usuario compartir el contenido almacenado en la aplicación con distintas aplicaciones de la plataforma.
4. Brindar la oportunidad al usuario de catalogar el contenido almacenado, para su fácil acceso mediante una herramienta de búsqueda y filtrado.

IV.3 Alcance de la aplicación

La aplicación, titulada **tagR** y que estará disponible en la tienda de aplicaciones de Google Play Store es una aplicación móvil orientada a consumir el contenido almacenado en etiquetas NFC. Sin embargo, el usuario también podrá generar contenido que desee almacenar en etiquetas NFC.

Por otra parte la aplicación invita al usuario a almacenar el contenido leído para consultarlo posteriormente, pudiendo este categorizarlo mediante etiquetas para facilitar el proceso de búsqueda.

La aplicación brinda la posibilidad de compartir el contenido almacenado con otras aplicaciones instaladas en el dispositivo tales como clientes de mensajería vía *SMS* y/o *Web*, navegadores *Web*, gestores de notas, procesadores de imágenes, etc.

IV.4 Plataforma

En lo referente a la plataforma de despliegue, está previsto que la aplicación este a disposición de los usuarios a través de la tienda de aplicaciones de Google, *Play Store*, para los dispositivos móviles que cumplan con los siguientes requerimientos:

- Sistema Operativo Android versión 4.0 (Ice Cream Sandwich) o superior
- Conectividad NFC incorporada
- Pantalla de 4" de diagonal o similar

Para el desarrollo de la aplicación se utilizaron los siguientes componentes de hardware y software:

- **Computador:** Dell® Inspiron™ 3847
 - Hardware
 - Procesador: Intel® Core™ i5-4460
 - Memoria RAM: 8 Gb
 - Software
 - Sistema Operativo: Microsoft® Windows™ 8.1
 - Entorno de desarrollo: Eclipse versión (4.3.1) con el *plug-in Android Developers Tools* (ADT) versión 23.0.2
- **Teléfono:** Samsung Galaxy SIII (Samsung I9300)
 - Hardware:
 - Procesador: Exynos 4412 Quad Core
 - Memoria RAM: 1 GB
 - Software:
 - Sistema Operativo: Android versión 4.3 (Jelly Bean)

- **Teléfono:** Google Nexus 4 (LG E960)
 - Hardware:
 - Procesador: Qualcomm APQ8064 Snapdragon Quad-core
 - Memoria RAM: 2 GB
 - Software:
 - Sistema Operativo: Android versión 5.0.1 (Lollipop)

- **Etiqueta NFC:** NTAG216 (2 Unidades)
 - Hardware:
 - Tipo de Circuito Integrado: NTAG216
 - Fabricante: NXP Semiconductors
 - Memoria: 888 bytes
 - Clasificación según el *NFC Forum*: Tipo I.²

- **Etiqueta NFC:** Topaz-512 (2 Unidades)
 - Hardware:
 - Tipo de Circuito Integrado: Topaz-512
 - Fabricante: Broadcom Corporation
 - Memoria: 512 bytes
 - Clasificación según el *NFC Forum*: Tipo I.

- **Etiqueta NFC:** MIFARE Classic (2 Unidades)
 - Hardware:
 - Tipo de Circuito Integrado: MIFARE Classic
 - Fabricante: NXP Semiconductors
 - Memoria: 1b kbyte
 - Clasificación según el *NFC Forum*: Tipo II.

Ambos teléfonos fueron utilizados en conjunto con las etiquetas NFC para probar las funcionalidades de comunicación de la aplicación.

² Ver tabla 2-1. Comparación de los tipos de etiquetas NFC.

IV.5 Descripción de la aplicación

En esta sección se describen a detalle los diferentes elementos de la aplicación, así como sus usuarios, los requerimientos levantados, las funcionalidades implementadas, las principales interfaces y las pruebas realizadas junto con sus resultados.

IV.5.1 Usuarios a los que está dirigida

La aplicación estará dirigida a los usuarios de dispositivos móviles con sistema operativo Android compatibles con la tecnología NFC, interesados en compartir contenido multimedia rápida y fácilmente. Entre estos usuarios se consideraron dos perfiles de usuario a lo largo del diseño de la aplicación:

- Usuario Generador de Contenidos: este usuario crea contenido que será almacenado en etiquetas NFC, el cual será almacenado automáticamente para una posible próxima escritura.
- Usuario Consumidor de Contenido: este usuario lee el contenido almacenado en etiquetas NFC, el cual podrá ser almacenado para su posterior revisión.

IV.5.2 Principales requerimientos funcionales

Los principales requerimientos identificados para la aplicación son los siguientes:

- Almacenar diversos tipos de información en etiquetas NFC.
- Recuperar el contenido almacenado en las etiquetas NFC.
- Transmitir URLs, imágenes, textos, ubicaciones geográficas y variado contenido multimedia desde una etiqueta NFC a un celular con sistema operativo Android.
- Recibir un mensaje de texto vía SMS con información configurable a partir del contacto realizado entre un teléfono celular con sistema operativo Android.
- Registrar en una base de datos la interacción entre los teléfonos celulares y las etiquetas NFC.

IV.5.3 Principales requerimientos no funcionales

- **Escalabilidad:** La aplicación deberá ser escalable, siguiendo un esquema modular que facilite integrar nuevos módulos a los desarrollados y reutilizar servicios o funcionalidades ya desarrollados.
- **Usabilidad:** La aplicación deberá seguir los patrones de interacción y los lineamientos de diseño para la interfaz de usuario definidos y recomendados por Android con el objetivo de garantizar la mejor experiencia de uso.
- **Disponibilidad:** El contenido almacenado en la aplicación será disponible incluso si el dispositivo no está conectado a una red de datos.
- **Accesibilidad:** La aplicación deberá ofrecer soporte para los leguajes español e inglés.

IV.6 Desarrollo de la aplicación

IV.6.1 Funcionalidades desarrolladas

La aplicación está estructurada lógica y visualmente en cuatro módulos, los cuales agrupan todas las funcionalidades desarrolladas basadas en los requerimientos. La estructura es listada a continuación:

- 1. Lectura del contenido almacenado en una etiqueta NFC:** Permite realizar la lectura y visualizar los datos almacenados en una etiqueta NFC. Este módulo además incluye las funcionalidades de:
 - a. Almacenar el contenido leído:** Guardar el contenido obtenido en la base de datos de la aplicación.
 - b. Ejecutar una aplicación asociada al contenido:** En caso de que el contenido leído este asociado a una aplicación en particular y ésta esté instalada en el dispositivo que actúa como lector, se ofrece la opción de ejecutar la aplicación relacionada para visualizar el contenido de forma adecuada.
- 2. Escritura de contenido en etiqueta:** Agrupa las siguientes funcionalidades:
 - a. Crear nuevo contenido:** Presenta un formulario que permite crear contenido de alguno de los tipos definidos por la aplicación. El usuario puede almacenar el dispositivo en la aplicación antes de escribirlo en la etiqueta.

- b. **Reutilizar el contenido almacenado:** En caso de existir contenido almacenado en la aplicación, este es desplegado en la vista principal de Escritura de contenido para brindar al usuario la posibilidad de reutilizar alguno de los registros previos en una próxima escritura.

3. Gestión del contenido almacenado en la aplicación: En este módulo se integran las siguientes funcionalidades:

- a. **Visualizar el contenido almacenado:** Despliega una lista de los contenidos almacenados en la aplicación. Por cada elemento de esta lista se ofrecen las siguientes opciones:
 - i. **Editar el contenido seleccionado:** Permite editar el contenido almacenado en la aplicación, pudiendo modificar cualquiera de sus campos. Además desde la interfaz de edición es posible iniciar la transferencia del contenido hacia una etiqueta NFC.
 - ii. **Eliminar el contenido seleccionado:** Permite borrar el registro seleccionado de la base de datos de la aplicación.
 - iii. **Compartir el contenido seleccionado:** Permite compartir el contenido seleccionado con las aplicaciones compatibles instaladas en el dispositivo.
 - iv. **Agregar una etiqueta al contenido seleccionado:** Permite crear y agregar una o más etiquetas al registro seleccionado. Una vez creada una etiqueta está disponible para ser agregada a cualquier otro contenido almacenado en la aplicación. Una etiqueta puede ser eliminada o modificada en cualquier momento sin que tal acción afecte al contenido asociado.
 - v. **Generar un código QR:** Genera y despliega un código QR (véase Anexo A) que contiene la información asociada al contenido seleccionado. Una imagen del código generado podrá ser almacenado en el dispositivo. Además es posible compartir el código QR con las aplicaciones compatibles instaladas en el dispositivo en formato de texto y/o imagen.
- b. **Filtrar el contenido según su tipo:** Permite refinar la lista de contenidos según su tipo.
- c. **Filtrar el contenido según las etiquetas asociadas este:** Permite refinar la lista de contenidos según las etiquetas asociadas a los contenidos.

- d. **Buscar un contenido específico:** Permite realizar la búsqueda de un contenido a partir de una palabra o frase clave.

4. Utilidades o Extras: Este módulo incluye las siguientes funcionalidades:

- a. **Borrar el contenido almacenado en una etiqueta NFC:** Permite eliminar el contenido de almacenado en una etiqueta NFC, restaurándola a su estado original.
- b. **Respalidar el contenido almacenado en la aplicación:** Ofrece la posibilidad de generar un código QR que recoge todo el contenido almacenado en la aplicación. Este código puede ser guardado en el dispositivo móvil o compartido en con alguna aplicación compatible instalada en el mismo.
- c. **Restaurar contenido en la aplicación:** Brinda la oportunidad de decodificar un código QR que haya sido generado por la aplicación y almacenar la información obtenida en la aplicación en caso de que no exista.

IV.6.2 Diagramas de Casos de Uso

A continuación se presentan Diagramas de Casos de Uso en dos distintos niveles de detalle que reflejan la estructura descrita en la sección anterior.

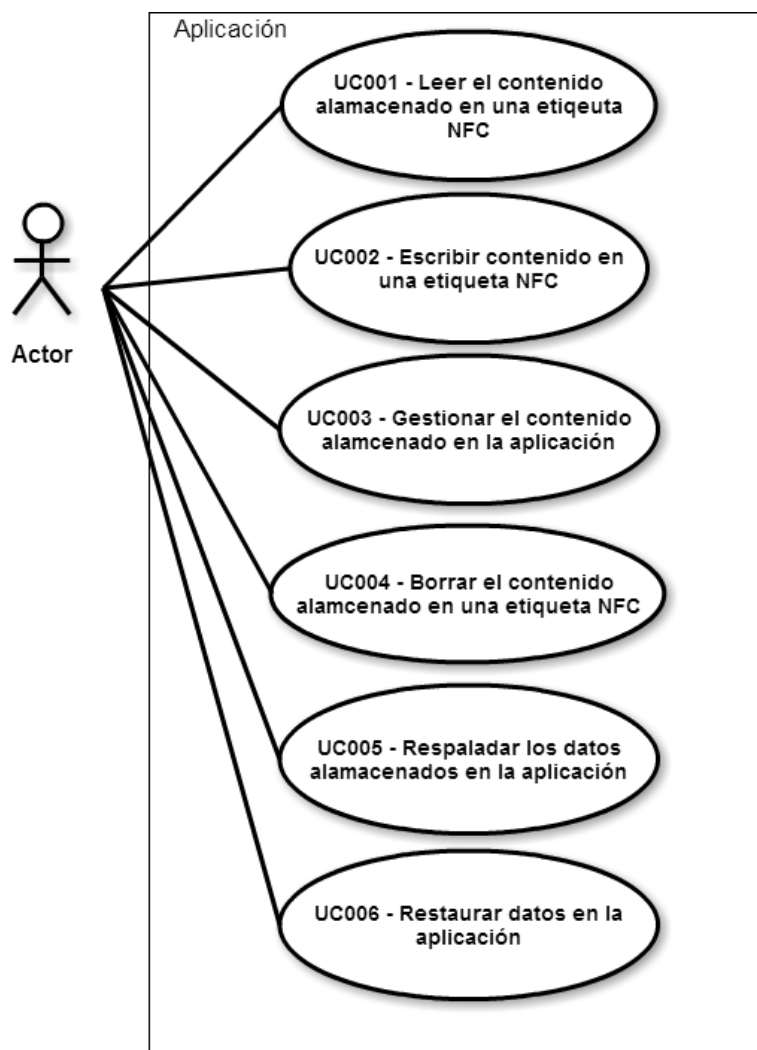


Figura 4-1. Diagrama de Casos de Uso, Nivel 0.
Fuente: Creada por el autor.

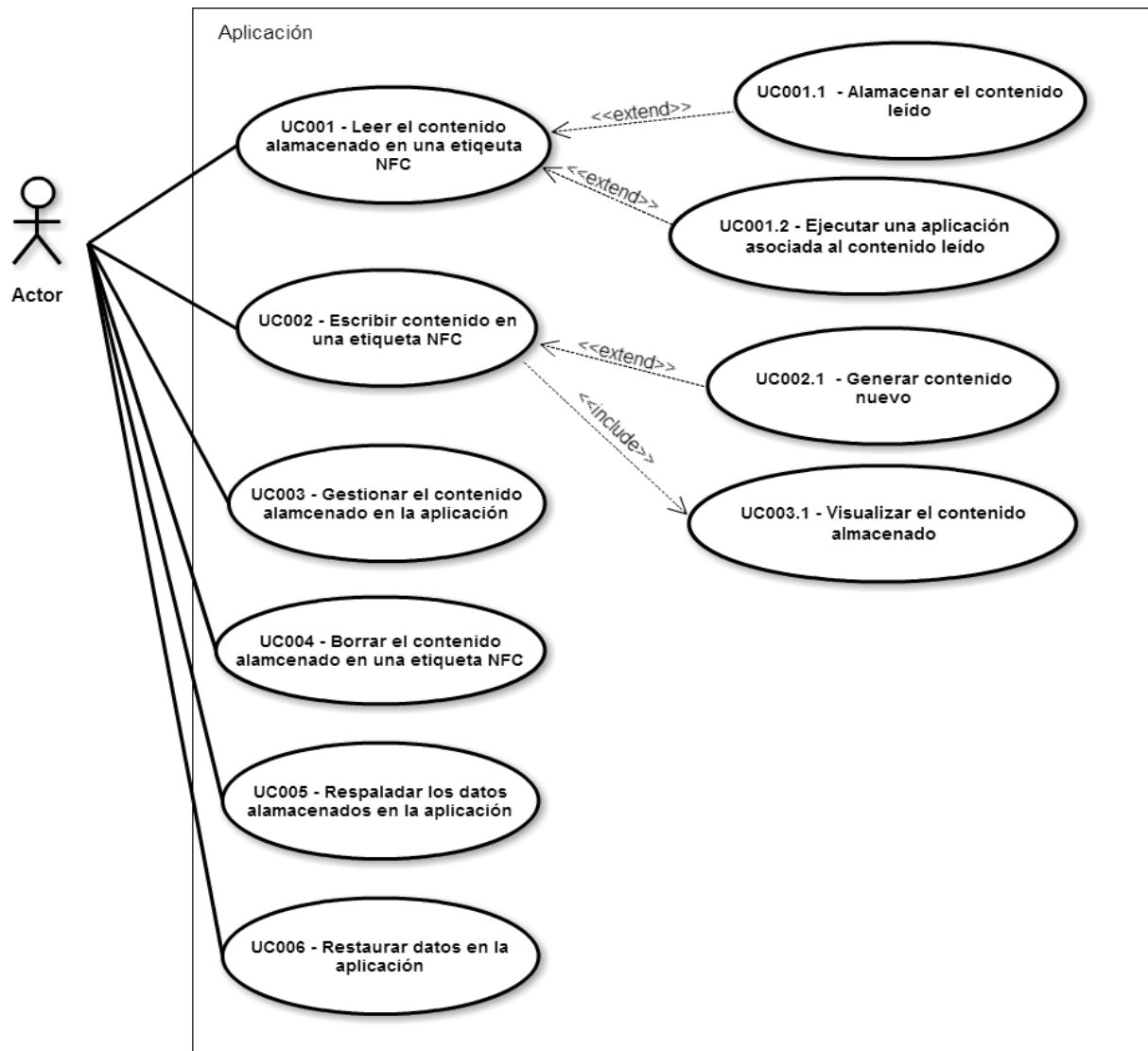


Figura 4-2. Diagrama de Casos de Uso, Nivel 1.
Fuente: Creada por el autor.

IV.6.3 Arquitectura de la aplicación

Los lineamientos de desarrollo de aplicaciones para el sistema operativo Android establecen que estas deben seguir el patrón Modelo-Vista-Controlador (MVC), patrón de arquitectura de aplicaciones de software para la implementación de interfaces de usuario que separa las representaciones internas de la información de las formas en la que esta es presentada al usuario.

Este patrón divide la estructura de la aplicación en tres partes:

- **Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación.
- **Vista:** Presenta el 'modelo' (información y lógica de la aplicación) en un formato adecuado para interactuar (interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.
- **Controlador:** Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información.

La implementación de este patrón en el sistema Android involucra los siguientes componentes que se corresponden con cada una de las partes definidas de la siguiente forma:

- **Manejador de Base de Datos:** Este componente se corresponde con el Modelo y permite la definición de entidades de datos y sus relaciones mediante el uso de tablas. El motor de Base de Datos por defecto en Android es SQLite, pero puede implementarse la comunicación con cualquier otra tecnología.
- **Layouts:** Se corresponde con la Vista y son archivos en formato XML que contiene los elementos visuales que conforman la interfaz con la que interactúa el usuario.
- **Activities:** Son generalmente clases en el lenguaje de programación Java y se corresponden con el Controlador. Definen el comportamiento de la aplicación en función de las acciones del usuario, estableciendo un vínculo entre la interfaz y el modelo de datos.

La aplicación desarrollada cumple con este patrón de arquitectura utilizando el motor de Base de Datos SQLite para la definición de Modelo de datos, cuyo diagrama es mostrado a continuación:

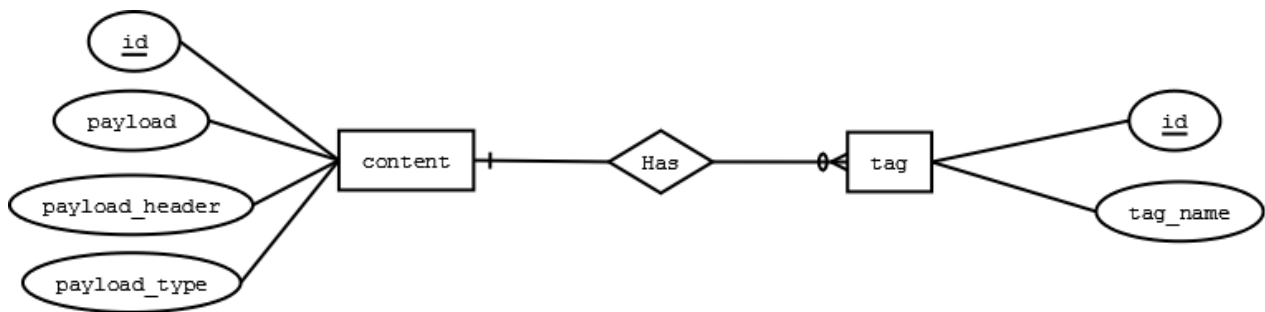


Figura 4-3. Diagrama Entidad – Relación del modelo de datos de la aplicación.

Fuente: Creada por el autor.

El diagrama Entidad – Relación mostrado en la Figura 4.3 satisface los requerimientos en cuanto a la gestión del contenido leído. Uno de los principales requerimientos es “Almacenar en el dispositivo móvil la información obtenida de la etiqueta NFC”, para lo cual se diseñó una entidad llamada **content** que permite almacenar toda la información necesaria del contenido leído para su posterior consulta y distribución, así como para facilitar su búsqueda y filtrado en los registros de la aplicación. Otro requerimiento considerado en el diseño del modelo de datos es “Clasificar el contenido mediante etiquetas o categorías”, por lo que se decidió incluir una entidad llamada **tag** que pudiese estar asociada con una o más entidades de tipo **content**, representado mediante la relación **Has**.

La traducción de este diagrama a un esquema de Base de Datos se puede apreciar en la siguiente figura.

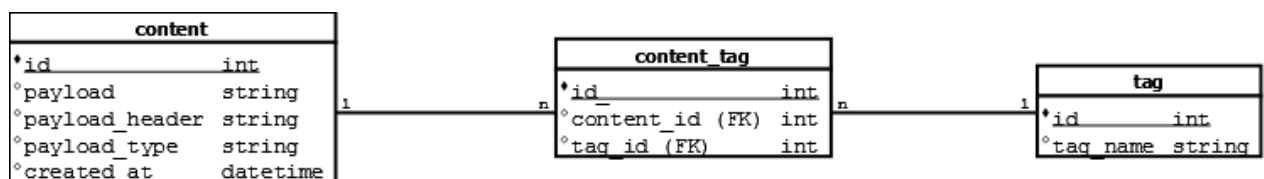


Figura 4-4. Diagrama del Modelo Relacional de datos de la aplicación.

Fuente: Creada por el autor.

Para el modelo de la base de datos, mostrado en la figura 4.4, las entidades **content** y **tag** se convierten en tablas relacionadas a través de la tabla **content_tag** que surge de la relación **Has** para manejar la cardinalidad muchos a muchos existente entre las dos entidades antes mencionadas.

IV.6.4 Implementación de las funcionalidades

Siguiendo la estructura listada en la sección 6.1, se describen a continuación los aspectos más resaltantes de las principales funcionalidades implementadas en la aplicación.

IV.6.4.1 Lectura del contenido almacenado en una etiqueta NFC

Para la implementación de esta característica en el *Activity* `ReadMain`, principal controlador del módulo de lectura, se utilizó un objeto de la clase `NfcAdapter` incluida en el paquete `android.NFC` del SDK de Android, la cual permite tener acceso al adaptador NFC presente en el dispositivo. Con este objeto se verifica el estado actual del adaptador y de ser posible se solicita su activación al sistema operativo para iniciar el proceso de comunicación con una etiqueta NFC.

De estar disponible el adaptador NFC del dispositivo se solicita al usuario mediante la vista generada por el *layout* `ReadMain` que aproxime la etiqueta NFC cuyo contenido desea leer a la parte posterior del equipo para iniciar la comunicación. El proceso de comunicación entre la aplicación y una etiqueta NFC es diagramado en la figura 4-5.

De ser exitosa la comunicación, la vista `ReadMain` es actualizada mostrando el contenido leído así como información técnica de la etiqueta que incluye capacidad de almacenamiento disponible, posibilidad de hacer la etiqueta de sólo lectura y una lista de tecnologías NFC soportadas, entre otras características de interés para el usuario escritor de contenido.

Si la etiqueta leída no está vacía y posee un registro compatible con la aplicación se ofrece la oportunidad al usuario de almacenar la información en el dispositivo. Para el proceso de guardado se inserta un nuevo registro en la tabla **content** (véase Figura 4-4) con el contenido leído y se informa al usuario el resultado de la operación.

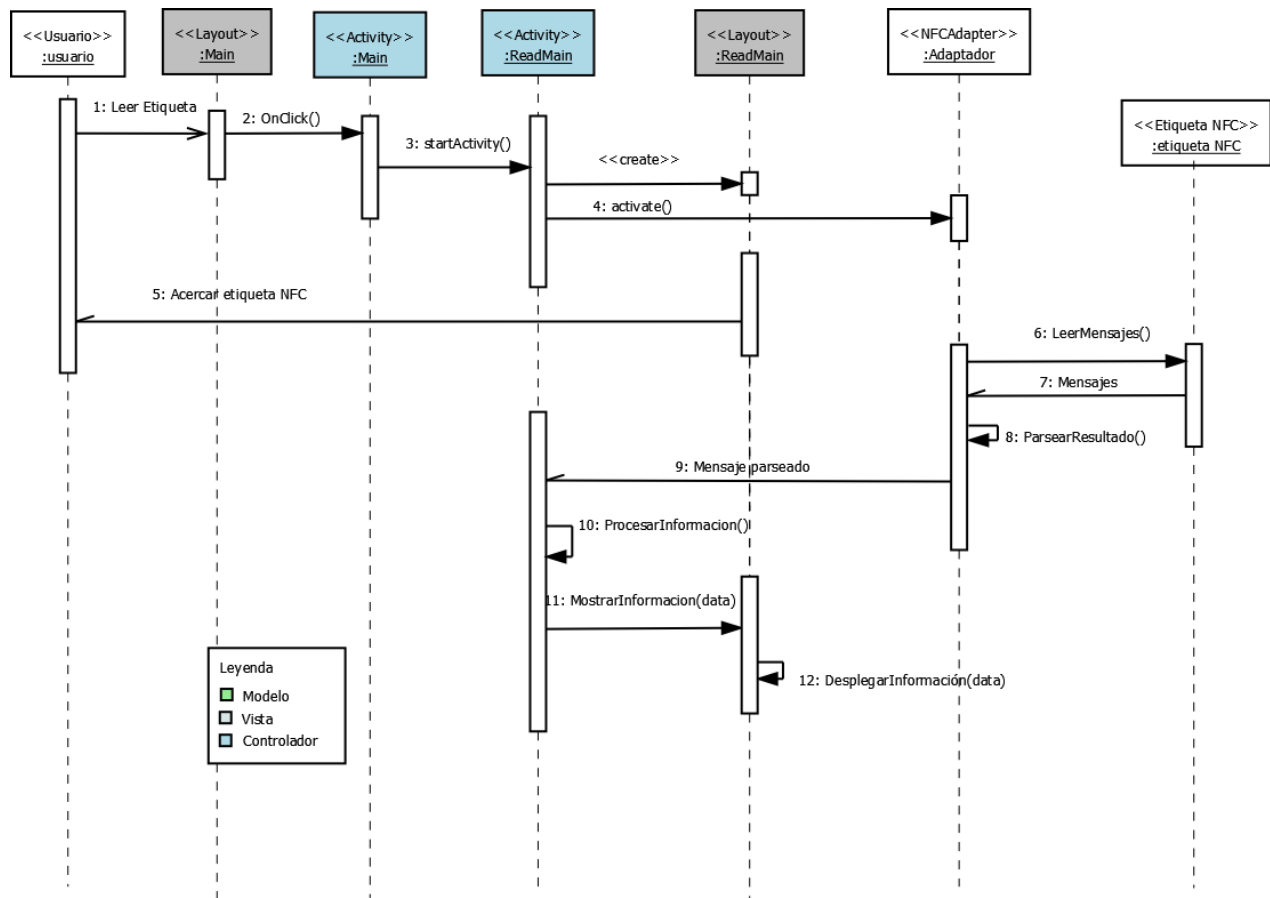


Figura 4-5. Diagrama de Secuencia de la funcionalidad "Lectura del contenido almacenado en una etiqueta NFC".
Fuente: Creada por el autor.

Como se observa en la figura 4-6, para el almacenamiento del contenido se utiliza un objeto *Helper* o asistente para acceder a la capa de datos de la aplicación. La clase asistente `customSQLiteOpenHelper` que hereda los métodos de la clase `SQLiteOpenHelper` definida en el paquete `android.database.sqlite` del SDK de Android, posee el método `insertContent()` que realiza la inserción de un nuevo registro de tipo **content** en la tabla homónima siempre y cuando el contenido a guardar no exista actualmente en la base de datos.

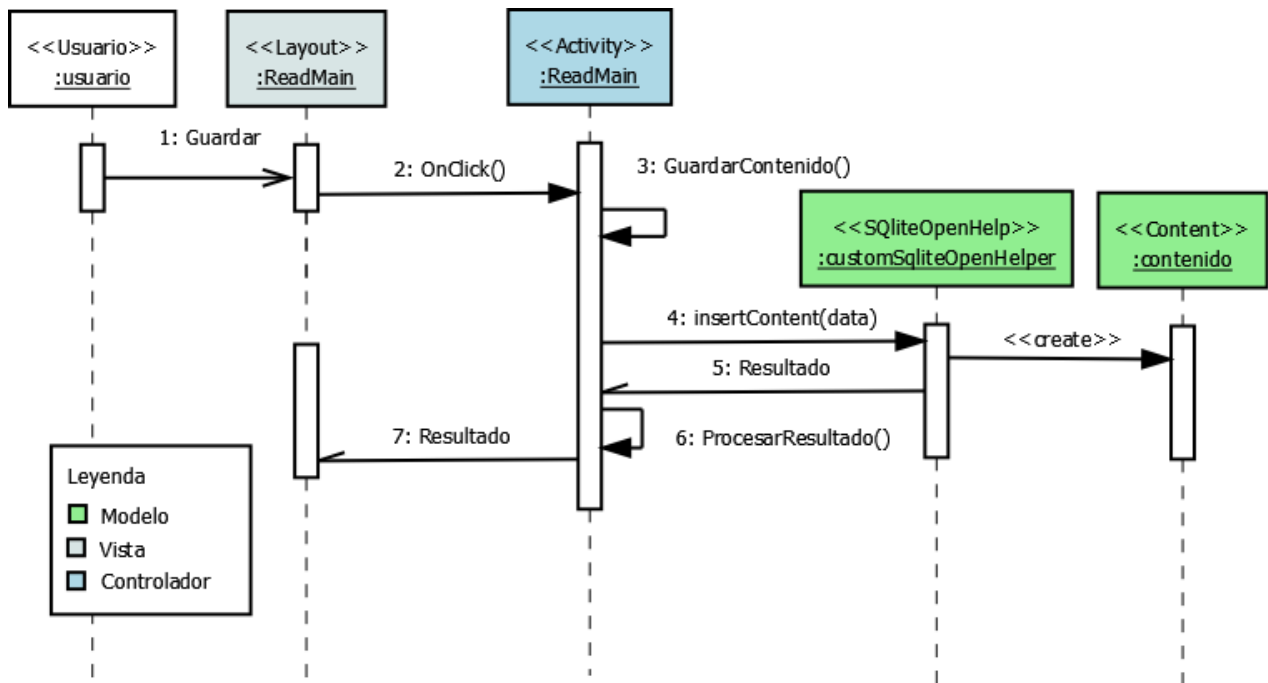


Figura 4-6. Diagrama de Secuencia del proceso de almacenamiento de contenido en la aplicación.
 Fuente: Creada por el autor.

Si en el mensaje obtenido de la etiqueta NFC existe un Registro de Aplicación Android o AAR (*Android Application Record*), tipo de registro NFC que asocia un contenido con una aplicación Android específica, la aplicación desarrollada ofrece la posibilidad de iniciar la aplicación indicada dentro del AAR si esta se encuentra instalada en el dispositivo, en caso contrario se redirigirá al usuario a la tienda de aplicaciones Google Play para iniciar su descarga.

Es importante destacar que la funcionalidad de lectura desarrollada es capaz de procesar un máximo de 2 registros por etiqueta, siendo obligatoriamente el segundo de ellos de tipo AAR, en contraste con la aplicación NFC Tools descrita en la sección de Antecedentes del Capítulo II que soporta la lectura de 2 o más registros sin importar su tipo. Esta decisión de diseño se basó en el funcionamiento de la aplicación NFC Writer (véase Capítulo II, sección II.4), considerando que las etiquetas NFC no están concebidas como unidades de almacenamiento masivo y que el uso de esta tecnología está enfocado en la rapidez del consumo y/o transferencia de la información.

IV.6.4.2 Escritura de contenido en una etiqueta NFC

Para llevar a cabo esta tarea el usuario dispone de dos opciones: Crear contenido o Utilizar el contenido almacenado en la aplicación. Ambas opciones son manejadas mediante el controlador o *Activity* `WriteMain`.

Para crear o generar nuevo contenido el usuario debe acceder a la interfaz de creación y elegir el tipo de contenido que desea crear de una lista de opciones por defecto. Para esta funcionalidad se diseñó una interfaz dinámica, a través del *layout* `CreateTagContent` y controlada por un *Activity* del mismo nombre, que varía en función del tipo de contenido seleccionado por el usuario, mostrando en cada caso un formulario ceñido a los campos asociados al tipo de contenido en cuestión.

Por ejemplo, si el usuario indica que el tipo de contenido a crear es "Ubicación Geográfica" el formulario mostrado, usando el *layout* o sub-vista `formGeo`, posee campos para introducir la longitud y la latitud del punto geográfico que se desea registrar así como también un botón que obtiene la ubicación actual del dispositivo utilizando las tecnologías de posicionamiento disponibles, para facilitar esta tarea.

En la vista principal del módulo de Escritura de contenido, generada por el *layout* `WriteMain`, el usuario dispone de una lista con los últimos 5 registros almacenados en la aplicación, pudiendo seleccionar cualquiera de ellos para utilizarlo en una nueva operación de escritura. De forma similar al escenario de creación, el usuario podrá editar los campos del contenido que desea escribir en la etiqueta antes de iniciar el proceso de transferencia de información.

En ambos escenarios, creación y reutilización, el usuario puede elegir si sólo desea almacenar el contenido en la aplicación, en cuyo caso el proceso es similar al mostrado en la figura 4-6, o si por el contrario desea escribirlo directamente en la etiqueta NFC. En este último caso el contenido es guardado automáticamente en la aplicación.

La transferencia de datos involucra la ejecución de los métodos `createContent()` y `saveContent()` pertenecientes a el *Activity* `CreateTagContent`. El primer método es llamado para validar que todos los campos del formulario hayan sido completados correctamente y de ser

así generar un `NdefRecord`, estructura de datos que será transferida a la etiqueta NFC y contendrá la información que se desea almacenar en el formato definido por el *NFC Forum*. El método `saveContent()` por su parte invoca al controlador `TransferContent` para iniciar el proceso real de escritura.

Para la transmisión del contenido a la etiqueta (véase figura 4-7) se utilizó, al igual que para la lectura de contenido, un objeto de la clase `NfcAdapter`. Una vez obtenido el acceso al adaptador NFC se inicia el proceso de escritura y al finalizar se le informa al usuario el resultado de la operación.

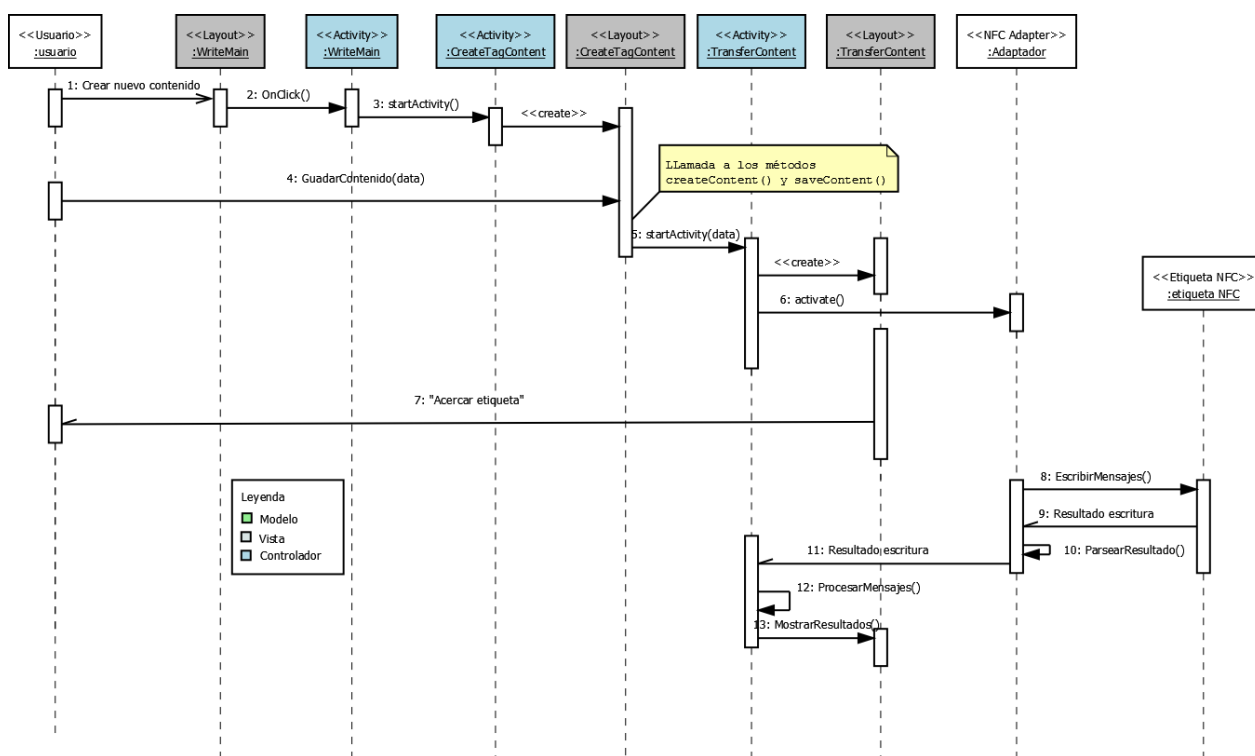


Figura 4-7. Diagrama de Secuencia de la funcionalidad "Escritura del contenido almacenado en una etiqueta NFC".
 Fuente: Creada por el autor.

IV.6.4.3 Gestión del contenido almacenado en la aplicación

Este módulo fue diseñado para que el usuario pueda manipular el contenido almacenado en la aplicación, por lo que en la vista principal de esta sección, presentada mediante el *layout* `TagsMain` y manejado por un controlador homónimo, es posible visualizar adecuadamente todos los contenidos almacenados en la base de datos, desplegados en forma de lista. Para obtener los registros que conforman la mencionada lista, se utiliza la función `getAllContents()` definida en la clase `customSQLiteOpenHelper`, la cual realiza una consulta sobre la tabla **content** que recupera todas filas existentes. Cada fila es procesada y la información que contiene es dispuesta en un *layout* `TagUIContent` asociado a un objeto del mismo nombre el cual es agregado a lista desplegada en la vista principal.

La lista de contenidos es manejada por un adaptador de listas o `ListAdapter` definido por el autor para controlar los eventos de pulsación sobre los elementos de la lista. Siguiendo los lineamientos de interfaz y experiencia de usuario definidos por Android y el comportamiento observado en las aplicaciones tomadas como referencia (véase Capítulo II, sección II.4) se implementaron procedimientos para responder a dos tipos de pulsaciones: corta y larga. Si el usuario pulsa sobre un elemento de la lista (pulsación corta) se muestra la interfaz de edición comentada en la sección IV.6.4.2 donde puede editar el contenido y actualizarlo en la base de datos y/o escribirlo en una etiqueta NFC.

En el caso de que el usuario presione un elemento de la lista durante unos pocos segundos (pulsación larga) se despliega un diálogo modal que contiene una lista de opciones aplicables a dicho contenido. Estas opciones son explicadas a continuación:

- **Eliminar:** Al pulsar sobre esta opción se muestra un diálogo que solicita al usuario confirmación sobre la decisión de eliminar el elemento presionado. De ser afirmativa la respuesta el elemento será eliminado de la base de datos llamando al método `deleteContent()` (véase figura 4-8) desde el `ListAdapter`, pasándole como parámetro el **id** (identificador clave primaria) del **content** que se desea eliminar.

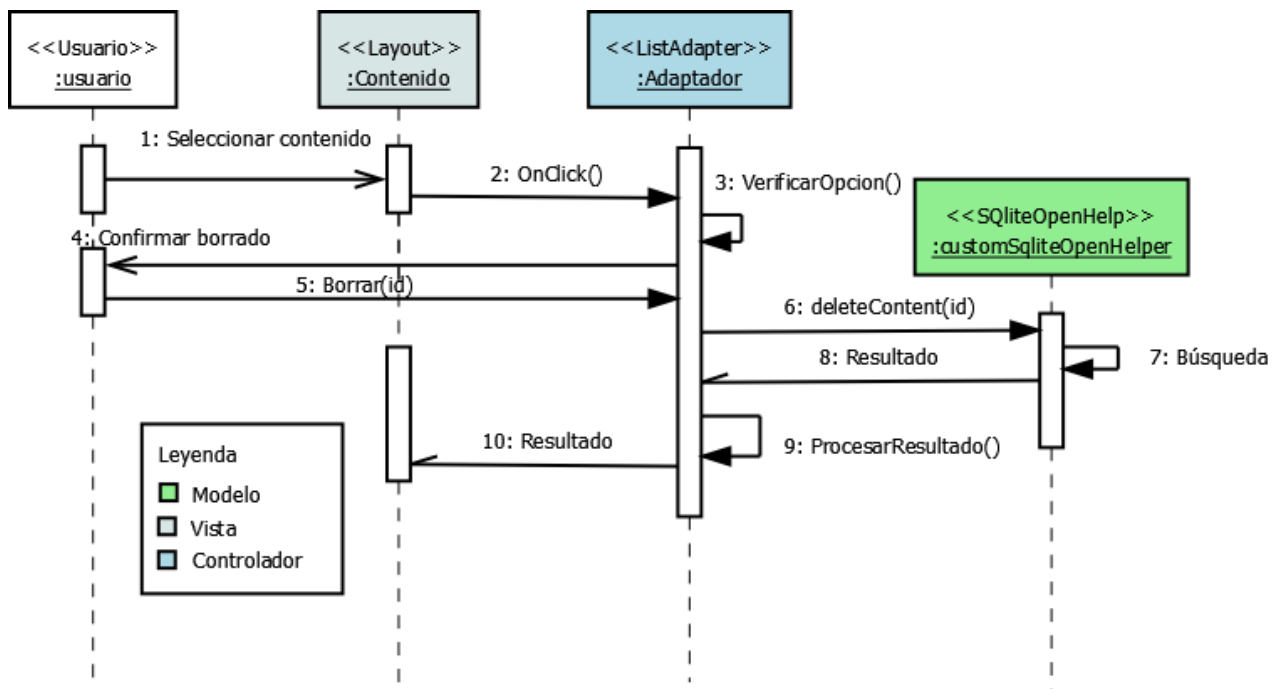


Figura 4-8. Diagrama de Secuencia de la funcionalidad "Eliminar contenido almacenado en la aplicación".

Fuente: Creada por el autor.

- **Compartir:** Si esta opción es seleccionada, la aplicación genera una cadena de texto que combina el contenido (**payload_header + payload**) y el tipo de dato del mismo (**payload_type**) presentes en el objeto `TagUIContent` y realiza una solicitud al sistema operativo para iniciar cualquier aplicación que sea capaz de procesar texto plano. En caso de existir más de una aplicación compatible, se despliega una lista con todas las aplicaciones disponibles permitiendo al usuario elegir con cual desea completar esta acción.
- **Agregar una etiqueta:** Esta funcionalidad permite asignar una o más etiquetas al contenido seleccionado. Al presionar esta opción aparece un diálogo (layout `addTag`) con un formulario donde el usuario puede escribir el nombre de la etiqueta que desea agregar. Al asignar una etiqueta un registro **tag** es creado y asociado al contenido seleccionado mediante la tabla **content_tag** (véase figura 4-4). Al crear una **tag** o etiqueta ésta queda disponible para ser asociada a cualquier otro contenido almacenado en la aplicación a través del diálogo principal de esta opción donde, de manera similar a la vista `TagsMain`, se despliega una lista con todas las etiquetas almacenadas.

La lista de etiquetas, controlada por el adaptador de listas `TagAdapter`, incluye un *checkbox* o casilla seleccionable por cada etiqueta existente que permite al usuario marcar y/o desmarcar para asignar y/o remover, respectivamente, la etiqueta seleccionada del contenido escogido. Manteniendo el patrón de interacción implementado en la lista de contenidos, al realizar una pulsación larga sobre el nombre de una etiqueta aparece un diálogo que ofrece las opciones de Editar el nombre de la etiqueta y Eliminar la etiqueta en cuestión. El funcionamiento de estas opciones es similar al ya explicado para la edición y eliminación de contenidos, por lo que no se profundizará en su descripción.

- **Generar un código QR:** Esta opción permite al usuario generar la imagen de un código QR que contiene una cadena de texto similar a la generada en la opción compartir mencionada anteriormente. Para la generación de la imagen del código QR se utilizó la biblioteca de código abierto `zxing`, que facilita el procesamiento de imágenes de códigos de barra multi formato de una y dos dimensiones. Al seleccionar esta opción se invoca al controlador o *Activity* `BackupData` que despliega el *layout* homónimo donde se presenta el contenido almacenado en texto plano junto con la imagen del código QR generada, ofreciendo las opciones de almacenar la imagen en el dispositivo y de enviarla a alguna aplicación compatible (véase el apartado compartir de la sección IV.4.6.3) en formato de texto plano o como imagen.

La lista de contenidos puede ser filtrada (véase figura 4-9) por dos parámetros: Tipo de contenido y Etiquetas asociadas a un contenido. Ambas opciones pueden ser invocadas independientemente al pulsar los botones destinados para tal fin ubicados en la región denominada `ActionBar` de la interfaz desplegada a través del *layout* `TagsMain`. Al pulsar en cualquiera de los dos botones se despliega un diálogo (*layout* `AddFilter`) con una lista de criterios (tipos de contenidos o etiquetas) seleccionables mediante *checkboxes*. Esta lista es controlada a través del `ListAdapter` `FilterKindAdapter` donde se define el comportamiento de la aplicación cuando el *checkbox* de un elemento de tipo `FilterKind` es marcado o desmarcado añadiendo o removiéndolo, según sea el caso, de un vector de criterios creado para realizar la operación de filtrado, la cual en ambos casos consiste en realizar una consulta a la base de datos mediante la llamada a la función `getTagUIContents()` del objeto `customSQLiteOpenHelper` que retorna un vector de objetos tipo `TagUIContent`, cuyos elementos son examinados para

obtener aquellos donde el atributo en cuestión (tipo de dato o etiqueta) coincide con los presentes en el vector de criterios mencionado anteriormente.

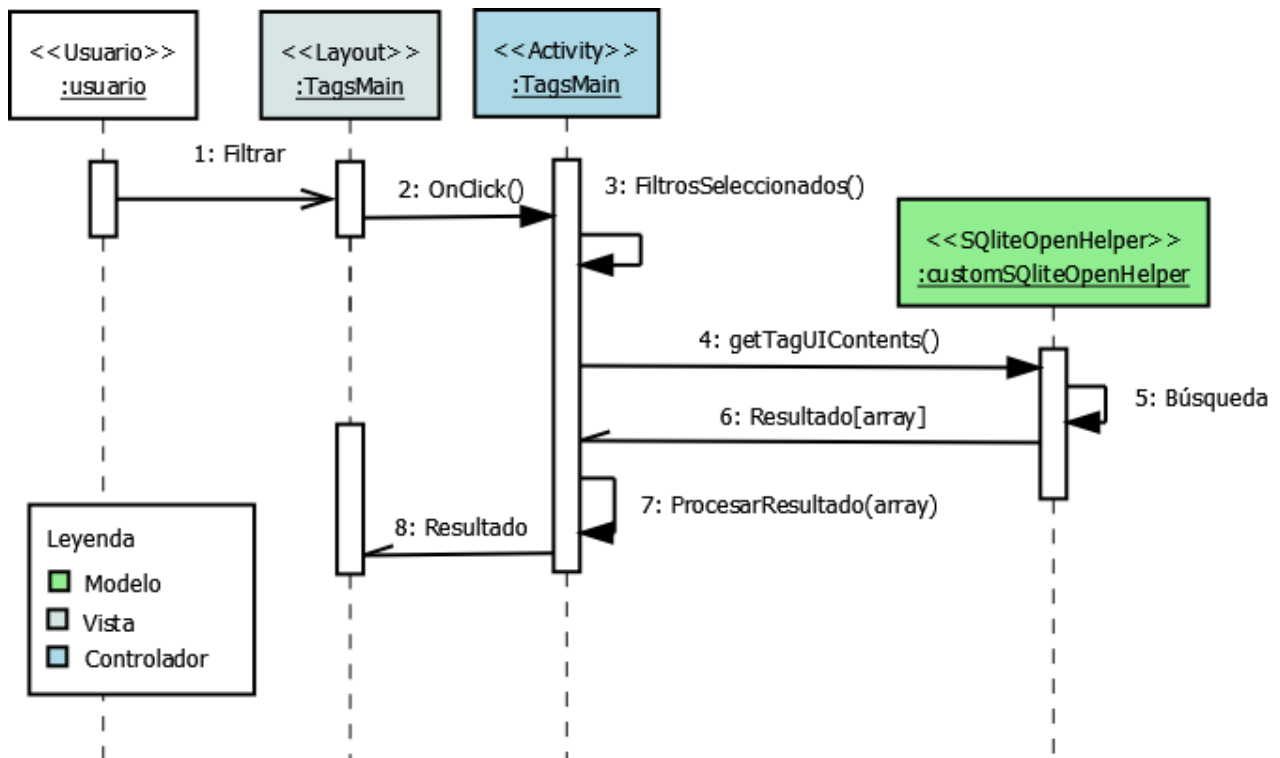


Figura 4-9. Diagrama de Secuencia de la funcionalidad "Filtrar el contenido almacenado en la aplicación".

Fuente: Creada por el autor.

Si bien la funcionalidad de filtrado fue implementada de forma independiente pero no por tanto es excluyente, por lo que si se ha aplicado el filtro A sobre la lista de registros original (sin filtro) es posible aplicar el filtro B sobre esta lista (filtrada por A) obteniendo el subconjunto deseado. Esto se logró examinando siempre la lista resultado desplegada en la interfaz **TagsMain** en lugar de acceder a la base de datos por cada invocación de un filtro.

Otra funcionalidad presente en este módulo de Gestión de contenido es Buscar un contenido utilizando una palabra o frase clave. De este modo se ofrece la posibilidad de expandir el criterio de búsqueda al texto del contenido (**payload**) involucrando a su vez el

tipo de dato (***payload_type***) y etiquetas (***tag_name***) asociadas a un contenido. A nivel visual la acción de búsqueda se incorporó mediante los componentes predefinidos de Android para esta acción con el fin de mantener el patrón de interacción utilizado por el usuario en otras aplicaciones de la plataforma, mientras que para el proceso de consulta de la base de datos se utilizó el objeto `customSQLiteOpenHelper` invocando el método `getContentBySearch()` que recibe como parámetro la palabra o frase ingresada por el usuario en el cuadro de búsqueda.

IV.6.4.4 Borrado del contenido existente en una etiqueta NFC

Esta funcionalidad forma parte del módulo denominado Extras, el cual incluye un conjunto de utilidades que aportan valor agregado a las funcionalidades de los módulos de escritura, lectura y gestión de contenido examinados anteriormente.

Mediante esta opción el usuario puede liberar el espacio ocupado en la memoria de una etiqueta NFC para restaurarla a su estado original. Esta función se logró creando un registro NDEF vacío que ocupe toda la sección de almacenamiento de la etiqueta y utilizando un proceso de escritura similar al visto en la sección 1.6.4.2.

IV.6.4.5 Respaldo del contenido almacenado en la aplicación

Esta característica brinda al usuario la facilidad de exportar o compartir todo el contenido almacenado en la base de datos de la aplicación a cualquier otra aplicación o usuario.

Al seleccionar esta opción de en la interfaz principal de la sección Extras el controlador `BackupData` accede a la capa de datos mediante un objeto de la clase `customSQLiteOpenHelper` para obtener todos los registros de la tabla ***content*** y generar una cadena de texto que contenga la información de cada contenido debidamente identificada. A partir de esta cadena de texto se produce un código QR mediante la ejecución de la función `encodeAsBitmap()` perteneciente a la clase `qrCodeEncoder` de la biblioteca `zxing` que retorna una imagen en formato BMP (*Bit MaP* o Mapa de Bits). Esta imagen es desplegada en la vista `BackupData` donde el usuario dispone de un botón que le permite Guardar la imagen generada y otro para que pueda Compartirla.

Si el usuario decide guardar la imagen del código QR generado, el `Activity BackupData` crea un archivo con extensión PNG (*Portable Network Graphics*) y se le notifica al usuario la ruta del mismo para que pueda consultarlo posteriormente. La ruta de almacenamiento escogida ha sido ubicada en el directorio de imágenes del sistema para garantizar un fácil acceso a los archivos generados por la aplicación.

Al elegir la opción Compartir, se despliega un diálogo que mediante un pequeño formulario con dos *radio buttons* o casillas seleccionables excluyentes, consulta al usuario si desea compartir el contenido como imagen (imagen del código QR) o como texto plano (cadena de texto generada). En función de la casilla marcada se muestra la lista de las aplicaciones disponibles en el dispositivo para intercambiar el contenido.

IV.6.4.6 Restauración de contenido en la aplicación

A manera de complemento de la característica descrita en la sección IV.6.4.5, se implementó la posibilidad de restaurar o importar un conjunto de registros a la aplicación a partir de la imagen de un código QR generado por la aplicación desarrollada.

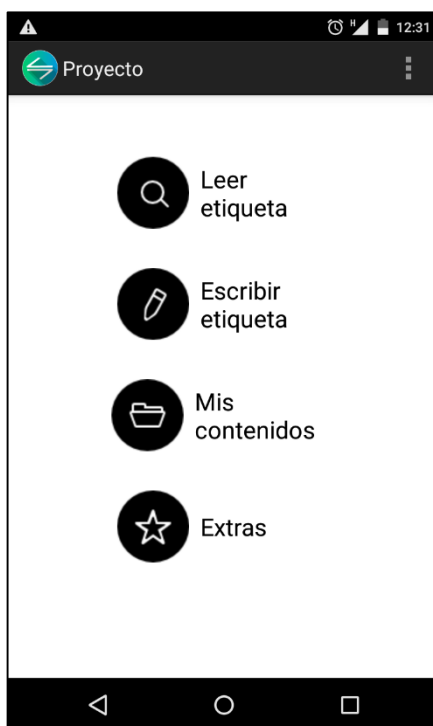
Esta funcionalidad ofrece al usuario la posibilidad de elegir entre Cargar una imagen almacenada en el dispositivo o Realizar una fotografía de un código QR. Si el usuario decide cargar una imagen se despliega el explorador de archivos del sistema para que el usuario seleccione la imagen con el código QR que desea procesar. El controlador `RestoreData` captura la imagen seleccionada por el usuario y mediante la ejecución de la función `decode()` se obtiene una cadena de texto con el contenido almacenado en el código. Si el contenido posee la cabecera definida por la aplicación durante la creación de un código QR se procede a extraer los registros de contenido que son mostrados al usuario a través de una lista similar a la observada en la interfaz principal de gestión (véase sección 1.6.4.3 de este capítulo), incluyendo también la opción de guardar en la aplicación los registros extraídos de la imagen que no existan en la base de datos.

En caso de que la opción elegida sea realizar una fotografía de un código QR, el controlador de esta funcionalidad invoca la aplicación de la cámara del sistema operativo y permite al usuario realizar la fotografía. Una vez el usuario está conforme con la fotografía

capturada, el proceso de la imagen obtenida es igual al descrito en el escenario de carga de una imagen almacenada en el dispositivo.

IV.6.5 Principales interfaces de la aplicación

El punto de entrada de la aplicación es la pantalla principal, con la siguiente disposición de elementos:



*Figura 4-10. Interfaz principal de la aplicación.
Fuente: Creada por el autor.*

Cada uno de los botones dispuestos en esta interfaz permite acceder al módulo de funciones indicado en el texto del mismo.

IV.6.5.1 Interfaces del módulo de lectura del contenido almacenado en una etiqueta NFC

Al acceder a este módulo se presenta al usuario un dialogo informativo indicándole que acerque un etiqueta NFC para iniciar la lectura del contenido.



Figura 4-11. Interfaz principal del módulo de lectura.
Fuente: Creada por el autor.

El contenido obtenido de la etiqueta es mostrado en la interfaz de la siguiente manera:



Figura 4-12. Interfaz de despliegue del contenido leído.
Fuente: Creada por el autor.

En la parte superior de la pantalla se muestra el contenido extraído de la etiqueta junto con el botón “Guardar”, que permite al usuario almacenar el contenido en la base de datos de la aplicación. A cada tipo de contenido soportado por la aplicación se le ha asociado un icono para facilitar su identificación en las demás interfaces.

Bajo el contenido leído, se despliega una lista con las características técnicas de la etiqueta NFC procesada. Cada característica ha sido asociada a un icono para facilitar su ubicación en la lista.



Figura 4-12.b. Interfaz de despliegue del contenido leído.
Fuente: Creada por el autor.

Al guardar el contenido leído el usuario es dirigido a la interfaz de resultados, donde se muestra el resultado de la operación y se indica el contenido que ha sido almacenado permitiéndole regresar a la interfaz principal mediante el botón “Listo”.



Figura 4-13. Interfaz de despliegue del resultado de almacenamiento de contenido.
Fuente: Creada por el autor.

IV.6.5.2 Interfaces del módulo de lectura de contenido almacenado en una etiqueta NFC

Al acceder a este módulo, el usuario puede escoger entre crear contenido o utilizar algún contenido almacenado en la aplicación.

Al pulsar en el botón “Crear nuevo contenido” se inicia la interfaz de creación y edición de contenido. Si desea utilizar contenido existente en la aplicación basta con presionar cualquiera de los elementos listados justo debajo del texto “Contenido reciente”, que de forma similar al botón “Crear nuevo contenido” despliega la interfaz de edición (véase figuras 4-15.b y 4-15.c) precargada con el contenido seleccionado.

La interfaz principal del módulo de escritura es mostrada a continuación.



Figura 4-14. Interfaz principal del módulo de escritura de contenido.
Fuente: Creada por el autor.

De existir cinco o más registros de contenido en la base de datos de la aplicación, se muestra el botón “Mostrar más” que permite acceder a la interfaz principal de gestión de contenido. En caso de no existir ningún elemento almacenado en la aplicación, se notifica al mensaje mediante un mensaje.

La interfaz de creación y edición de contenido posee un formulario dinámico (véase figura 4-15.a) que varía el número y tipo de campos en función del tipo de dato seleccionado en la lista desplegable ubicada en la parte superior izquierda.

El tamaño del final del mensaje es indicado al usuario mediante un texto que se actualiza dinámicamente y en tiempo real de acuerdo a la cantidad de caracteres existentes en los campos del formulario, logrando así que el usuario conozca cuantos bytes ocupará el contenido generado antes de iniciar el proceso de escritura.

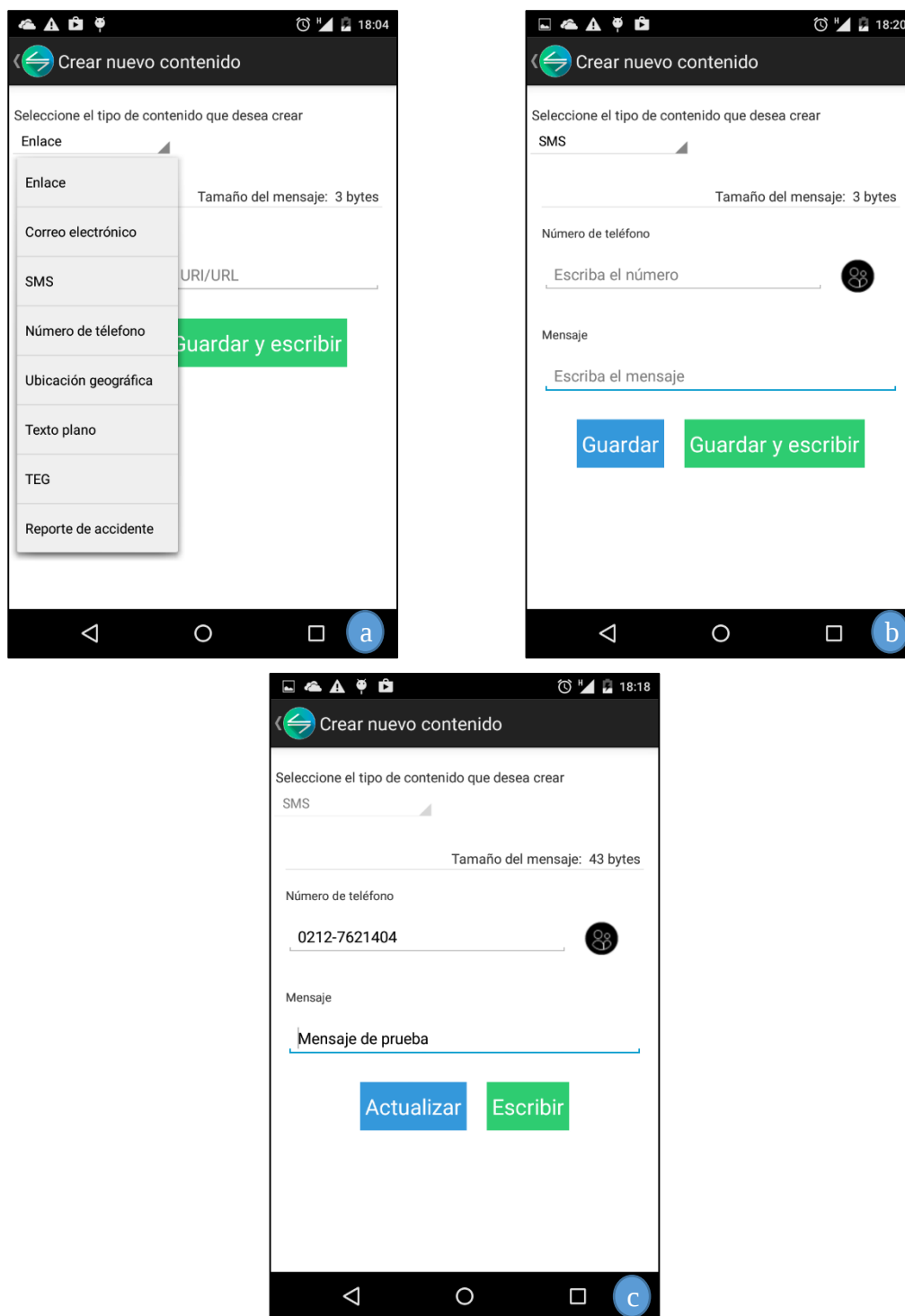


Figura 4-15. Interfaz de creación y edición de contenido.
a: Detalle de la lista desplegable. b: Formulario para la creación de un mensaje de texto.
c: Formulario para la edición de un mensaje de texto.
Fuente: Creada por el autor.

Si el usuario elige Guardar el contenido es dirigido a la interfaz vista en la figura 4-13, si en cambio decide Guardar y escribir en la etiqueta comienza el proceso de transferencia de contenido del dispositivo hacia la etiqueta y la interfaz mostrada en la figura 4-16.a es desplegada. Una vez terminado el proceso se actualiza la interfaz informando el resultado de la operación, indicando el contenido previo y el contenido actual de la etiqueta NFC involucrada, como se observa en la figura 4-16.b.

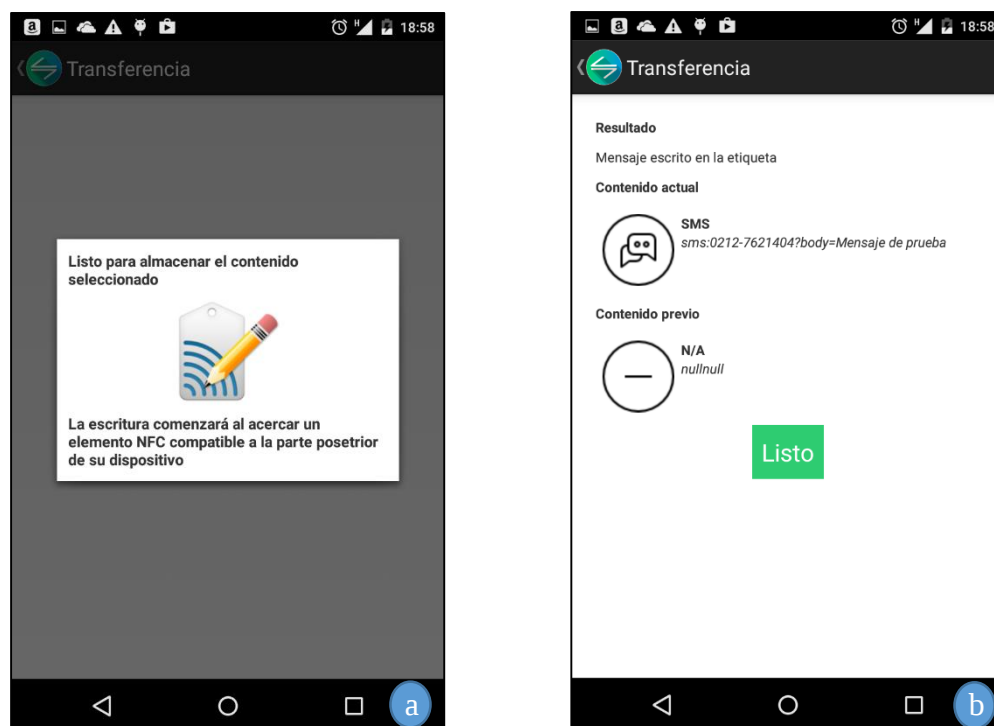


Figura 4-16. Interfaz de transferencia de contenido.
a: Inicio de la transferencia de contenido. b: Resultado de la transferencia.
Fuente: Creada por el autor.

IV.6.5.3 Interfaces del módulo de gestión de contenido almacenado en la aplicación

La interfaz principal de este módulo presenta una lista de todos los contenidos almacenados en la base de datos de la aplicación en orden descendente, como se observa en la figura 4-17.

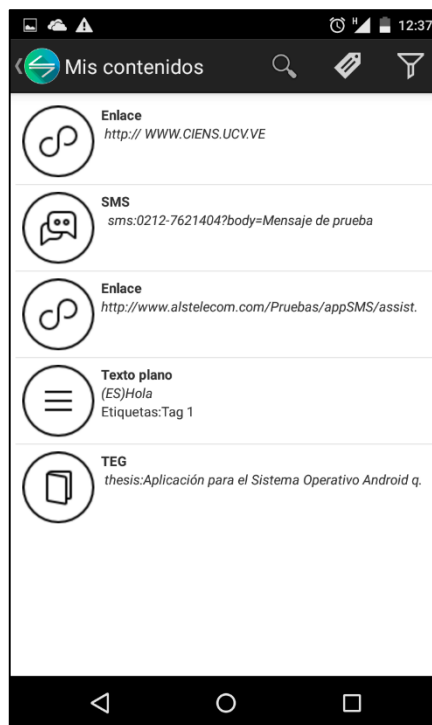


Figura 4-17. Interfaz principal del módulo de escritura de contenido.
Fuente: Creada por el autor.

Si uno de los elementos listados es presionado brevemente se despliega el formulario de edición de contenido (véase figura 4-15.b). Si el elemento de la lista es presionado sostenidamente durante unos segundos se presenta al usuario un menú contextual, a manera de diálogo modal como se aprecia en la figura 4-18.a, con las opciones aplicables a un contenido.

Al presionar cualquiera de las opciones presentadas se muestra un conjunto de diálogos interactivos que guían al usuario en el proceso de aplicar la acción seleccionada. En la figura 4-18.b se muestra el diálogo de consulta desplegado al presionar sobre la opción Eliminar.

Los apartados c y d de la citada figura permiten observar las interfaces generadas al seleccionar las opciones Compartir y Generar código QR respectivamente.

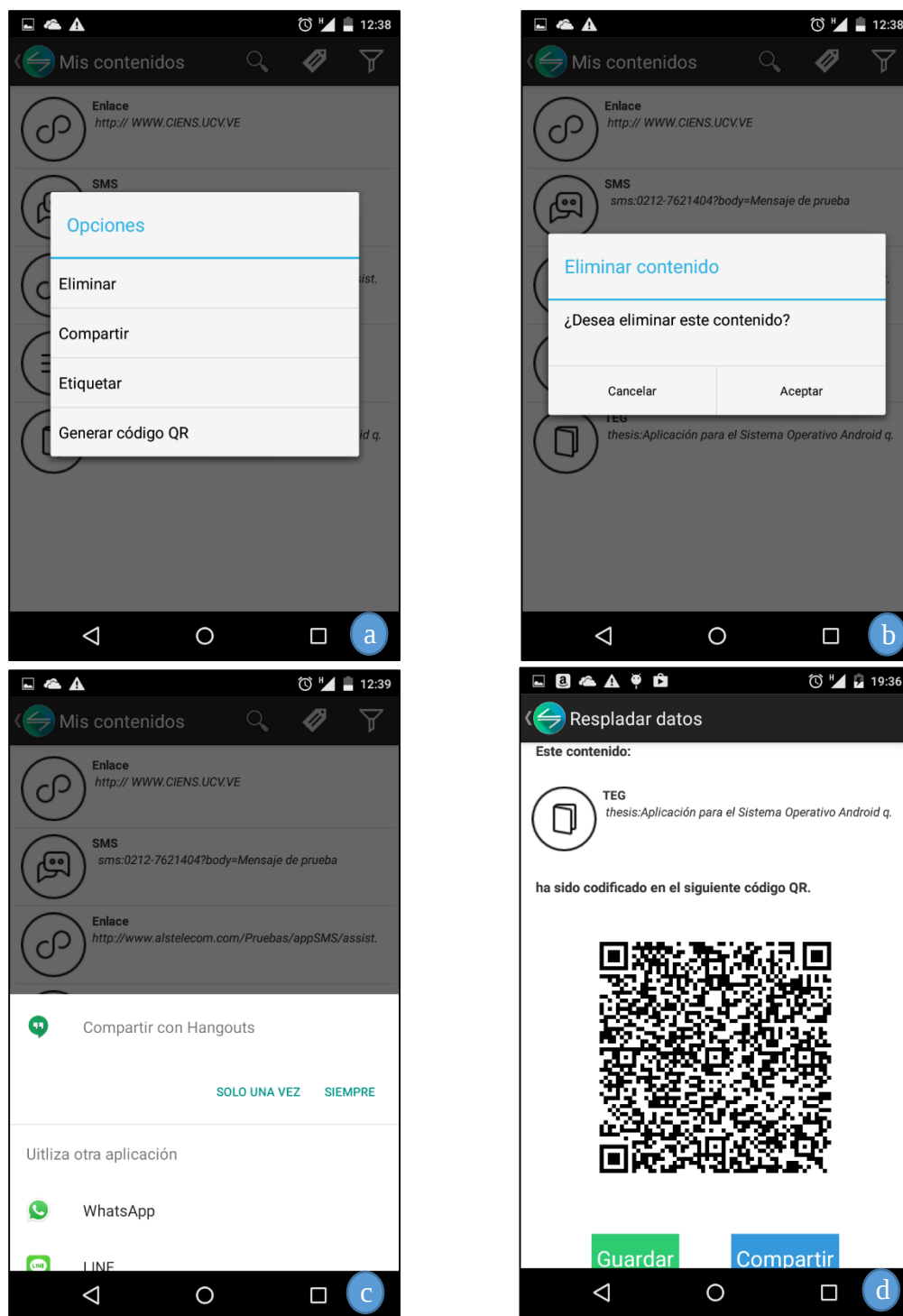


Figura 4-18. Interfaces con funcionalidades para el contenido.
a: Opciones aplicables a un contenido. b: Diálogo de la opción Cancelar.
c: Diálogo de la opción compartir. d: Interfaz de la opción Generar código QR.
Fuente: Creada por el autor.

Es posible realizar un filtrado de los registros por el tipo de contenido y por las etiquetas asociadas a este, además de poder realizar una búsqueda global incluyendo ambos criterios al presionar los botones con el ícono de lupa para iniciar una búsqueda, el ícono de la etiqueta para iniciar el filtrado por etiquetas y el ícono de filtro para iniciar el filtrado por tipo; ubicados en la región superior de la interfaz vista en la figura 4-17. Las interfaces de estas funcionalidades son mostradas en la figura 4-19.

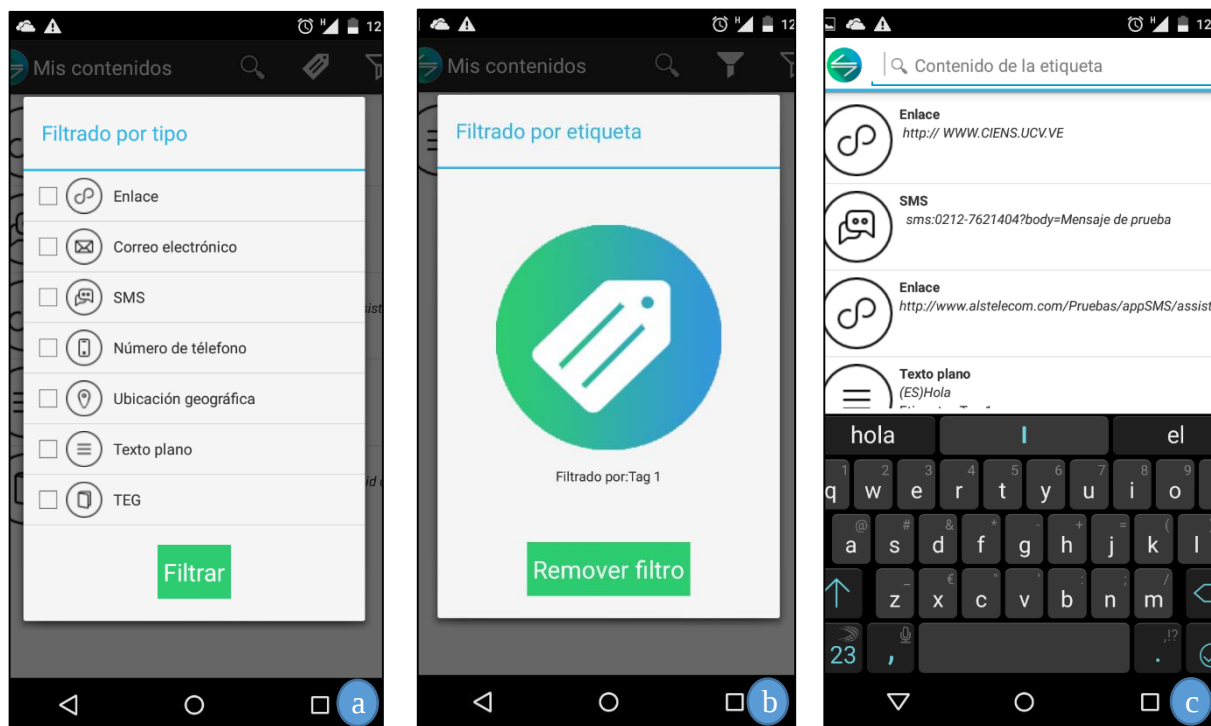


Figura 4-19. Interfaces de las opciones de filtrado.
a: Interfaz de filtrado por tipo. b: Interfaz de filtrado por etiqueta (aplicado).
c: Interfaz de barra de búsqueda.
Fuente: Creada por el autor.

IV.6.5.4 Interfaces del módulo de funcionalidades Extras

La interfaz principal de este módulo (figura 4-20.a) es similar a la interfaz principal de la aplicación para mantener consistencia en el diseño. Las interfaces del sub-módulo Borrar etiqueta poseen gran similitud con las observadas en el módulo de lectura de contenido.

Las interfaces principales de las funcionalidades Respalidar y Restaurar datos son mostradas en los apartados b y c, respectivamente, de la figura 4-2.

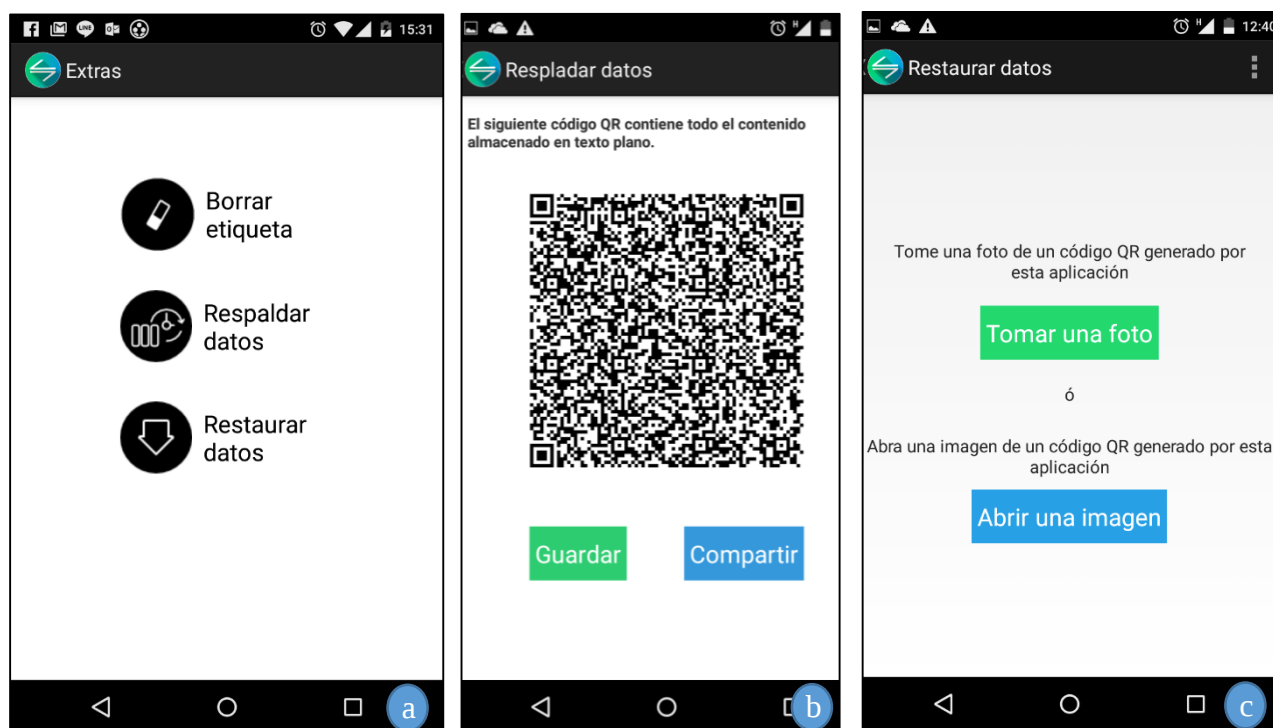


Figura 4-20. Interfaces de funcionalidades Extras.
a: Interfaz principal de módulo Extras. b: Interfaz principal del módulo Respalidar datos.
c: Interfaz principal del módulo Restaurar datos.
Fuente: Creada por el autor.

IV.6.6 Pruebas de la aplicación

IV.6.6.1 Pruebas unitarias

Siguiendo el modelo de Desarrollo Incremental e Iterativo, al finalizar cada iteración se realizaron pruebas del funcionamiento de los requerimientos realizados durante la misma, así como también pruebas de integración con las funcionalidades existentes. En caso de que alguna de las características no se comportara de la manera esperada, se realizaba un análisis de las causas del comportamiento erróneo y posteriormente se aplicaban las modificaciones pertinentes.

Las principales características sometidas a prueba en el módulo de lectura fueron la identificación de los distintos tipos de contenido soportados por la aplicación, la identificación de distintos modelos de etiquetas NFC y el manejo de excepciones en casos como lectura de una etiqueta vacía o intento de leer el contenido de una etiqueta cuando los sensores NFC del dispositivo están desactivados.

En el caso del módulo de escritura de contenido se verificó el comportamiento de los botones del formulario de creación y edición de contenido en el caso de que alguno de los campos estuviese vacío al momento de iniciar el almacenamiento (tanto en la base de datos de la aplicación como en la etiqueta NFC) del contenido. Se comprobó el funcionamiento del indicador del tamaño en bytes del mensaje a guardar para cada uno de los tipos de contenido soportados por la aplicación y el resultado obtenido coincide con el comportamiento observado en las aplicaciones utilizadas como referencia.

El módulo de gestión de contenido posee un nivel considerable de integración con los módulos de lectura, escritura y utilidades o funcionalidades extras. Se verificó que la funcionalidad de almacenamiento de contenido en la aplicación presente en distintas secciones de los otros tres módulos, además de guardar efectivamente el contenido indicado, no genere redundancia en la base de datos de la aplicación. Se validó el correcto despliegue de los registros almacenados en la aplicación, tanto en las listas generales como en las listas generadas como resultado de filtrados o búsquedas, y en caso de no existir ninguno se informe apropiadamente al usuario.

El correcto almacenamiento de las imágenes con códigos QR generadas por la aplicación, el reconocimiento de las imágenes generadas por la aplicación y el borrado correcto de las etiquetas NFC fueron las funcionalidades verificadas en el conjunto de pruebas aplicadas al módulo de utilidades.

IV.6.6.2 Pruebas de usabilidad

Durante la fase de pruebas de la aplicación se realizaron demostraciones y navegaciones guiadas de la aplicación a un grupo de 10 personas conformado por estudiantes de Ciencias de la Computación y personas ajenas al área, todos familiarizados con el funcionamiento del sistema operativo Android y las aplicaciones móviles. Para la demostración, en la mayoría de los casos se facilitó un dispositivo con la aplicación previamente instalada y en otros se les cedió el archivo instalable para que pudiesen probarlo desde un dispositivo propio. Mientras los usuarios navegaban por la aplicación se le instó a comentar de forma clara y sincera sus percepciones con respecto a los elementos de la interfaz y las funcionalidades implementadas. Finalizada la demostración y/o navegación de la aplicación se envió al usuario una encuesta (véase anexo C) vía correo electrónico, utilizando para su creación la herramienta *web* Formularios de Google o *Google Forms*, la cual constaba de 6 preguntas obligatorias de selección simple y una pregunta opcional donde se solicitaba al usuario que colocara cualquier comentario u opinión que no hubiese podido concretar en las preguntas previas.

Del análisis de los comentarios, consideraciones y opiniones obtenidas durante las pruebas se pudieron identificar ciertos aspectos de la interfaz y comportamiento de la aplicación que requerían modificaciones menores para lograr una mejor experiencia de uso.

Las modificaciones más resaltantes realizadas en función de las pruebas y sugerencias fueron:

- Incremento del tamaño de la fuente utilizada en los textos de algunas vistas.
- Mejora de la redacción de algunos textos en los diálogos informativos.
- Estandarización del tamaño de los íconos de los contenidos almacenados en la aplicación.
- Remplazo de algunos iconos que no transmitían la intención deseada.
- Mejoras en la transición y orden de aparición de algunas pantallas de la aplicación.

- Incorporación de funcionalidades extras para ampliar la utilidad de la aplicación, en especial funcionalidades para almacenar datos de la etiqueta NFC en códigos QR o recuperar información de utilidad para la aplicación desde éstos códigos.

La aplicación tuvo una buena recepción por parte del grupo de prueba cuyos integrantes la calificaron como “Muy intuitiva” y “bien diseñada”. De las respuestas obtenidas mediante la encuesta (véase anexo C, apartado Resultados de la encuesta) se pueden concluir los siguientes hechos:

- Es fácil determinar el objetivo de la aplicación.
- Las metáforas visuales aplicadas mediante iconos fueron entendidas fácilmente.
- La disposición de los elementos en la interfaz resultó adecuada para la mayoría de los usuarios.
- La transición entre las distintas vistas y elementos de la interfaz ocurren con la fluidez acostumbrada en el sistema operativo Android.

Resultados del Trabajo Especial de Grado

Como principal resultado del Trabajo Especial de Grado se obtuvo una versión de la aplicación apta para la distribución capaz de intercambiar contenido mediante el uso de la tecnología NFC entre dispositivos móviles Android y etiquetas NFC.

La aplicación desarrollada permite al usuario las siguientes operaciones: Lectura y almacenamiento en la aplicación del contenido escrito en una etiqueta NFC, Escritura de contenido (nuevo o existente en la aplicación) en una etiqueta NFC, Gestión del contenido almacenado en la aplicación que incluye las opciones de visualización, filtrado, búsqueda, edición, eliminación, clasificación y distribución del contenido a otras aplicaciones de Android; Borrado del contenido almacenado en un etiqueta NFC, Generación de un código QR (véase anexo A) a partir del contenido almacenado y Restauración del contenido en la aplicación mediante la captura de un código QR generado por la aplicación, cumpliendo así con los primeros 6 objetivos específicos enunciados en el Capítulo I de este documento, relacionados con el levantamiento de requerimientos y el desarrollo de la aplicación.

Basado en los resultados extraídos de las pruebas realizadas, cuyo análisis se realiza en al final del Capítulo IV, se puede afirmar que la aplicación cumple en gran medida con los lineamientos de diseño y experiencia de usuario definidos por Android para el desarrollo de aplicaciones móviles.

Dada la arquitectura modular del desarrollo, este puede ser fácilmente modificado en función de las necesidades de un cliente específico permitiendo agregar nuevas estructuras o modificar las existentes para admitir y procesar nuevos tipo de contenido.

Finalmente, y según lo planteado en el objetivo específico n°7, se elaboraron los documentos esenciales para el soporte y extensión de la aplicación en futuras oportunidades, que incluyen los diagramas de casos de uso en tres niveles de detalle indicando inclusiones y extensiones de las funcionalidades implementadas, los diagramas de secuencia de las principales funcionalidades, los diagramas Entidad-Relación y Modelo de la base de datos para reflejar la estructura de la capa de datos de la aplicación, así como los suficientes comentarios en el código fuente de la aplicación que faciliten la mantenibilidad y escalabilidad.

Por razones ajenas al autor la aplicación aún se encuentra en proceso de publicación en la tienda de aplicaciones de Google, sin embargo se espera poder colocarla a disposición de usuarios de la tecnología NFC.

Conclusiones, Recomendaciones y Trabajos Futuros

La iteración actual del desarrollo ofrece a los usuarios de la plataforma Android familiarizados con el uso de la tecnología NFC una herramienta para la lectura y escritura de contenido de interés para una gran variedad de aplicaciones o usos. La idea del cliente que solicitó el desarrollo además de ofrecer esta aplicación de forma masiva en el mercado de aplicaciones, es poder generar una versión personalizada para sus clientes actuales que les permita crear contenidos específicos e integrar la interacción con las etiquetas NFC con la plataforma existente de envíos SMS.

La metodología de desarrollo utilizada permitió abordar con facilidad los frecuentes cambios en la prioridad de los requerimientos, logrando desarrollar primero aquellas funcionalidades que aportaron mayor valor agregado al producto final. La arquitectura modular enmarcada en el principio de alta cohesión y bajo acoplamiento facilitó la evolución e integración de las funcionalidades sin que implicara un gran esfuerzo y/o mayor retraso en la planificación del proyecto.

Si bien las etiquetas NFC no fueron concebidas como unidades de almacenamiento masivo, la diversidad de su uso ha provocado escenarios donde la longitud del contenido que se desea almacenar excede del tamaño de la etiqueta afectando la experiencia de uso. Considerando el factor de la capacidad de almacenamiento se investigaron métodos para comprimir o codificar las cadenas de texto antes de escribirla en la etiqueta y poder guardar una mayor cantidad de datos. De los algoritmos de compresión analizados durante la investigación el más resaltante fue el algoritmo de Huffman (véase anexo B) pero aunque está diseñado para la codificación de cadenas de texto su funcionamiento no generó una disminución significativa de la cadena original en las pruebas realizadas, además, usar este método involucra incluir las claves para la decodificación junto con la cadena, lo que aumentaría la longitud del mensaje. Sin embargo, se logró incluir un acortador de URL para el caso de que el contenido sea de este tipo.

Para trabajos futuros a partir del desarrollo realizado en esta iteración se sugiere incluir la funcionalidad de protección de una etiqueta NFC contra escritura no autorizada, ya que aunque esta característica formaba parte de los requerimientos iniciales decidió postergarse

su desarrollo debido al poco conocimiento del tesista en el área de circuitos integrados, cuyos registros deben ser codificados para llevar a cabo esta funcionalidad, aunado a la reducida cantidad de etiquetas NFC provista para las pruebas.

Aunque la aplicación desarrollada trata de reunir los mejores elementos de las aplicaciones analizadas como antecedentes, la idea nunca fue crear una copia fiel de algún producto existente en el mercado. Sin embargo, sí se buscara crear una aplicación que se asemeje aún más a las líderes del sector podrían incluirse funcionalidades como la posibilidad de asociar una aplicación con un contenido o procesar otros tipos de contenido como tarjetas de contacto o *Smart Posters*.

REFERENCIAS BIBLIOGRÁFICAS Y DÍGITALES

- [1] Coskun V., Ok K., Ozdenizci B. (2013). *Professional: NFC Application Development for Android™*. [en línea] EEUU: Wrox. Disponible en: it-ebooks.info/read/2218/
- [2] Ortiz. S. *Near Field Communication*. [en línea] Asunción, Paraguay: Universidad Católica "Nuestra Señora de la Asunción". Disponible en: www.jeuazaru.com/docs/NFC.pdf
- [3] Ortiz L. (2012). *Diseño e implementación del prototipo de un dispositivo identificador de objetos de uso común para personas no videntes basado en la tecnología RFID*. [en línea] Ecuador: Universidad Politécnica Salesiana. Disponible en: <http://goo.gl/RWK1jt>
- [4] (2005) *Ventajas y desventajas del RFID*. Recuperado el 07 de agosto de 2014, de <http://goo.gl/jQd8Sc>
- [5] Chavarría D. (2011). *Tecnología de comunicación de campo cercano (NFC) y sus aplicaciones*. Costa Rica: Universidad de Costa Rica.
- [6] Tolsada Bris L. (2012). *Desarrollo de una aplicación de transferencia de ficheros basada en NFC y Bluetooth*. Madrid: Universidad Carlos III de Madrid.
- [7] Campa Ruiz A. (2011). *Desarrollo de una aplicación de pago a través de la tecnología NFC*. Madrid: Universidad Carlos III de Madrid.
- [8] Carignano M., Ferreyra, P. (2011). *Tecnología inalámbrica Near Field Communication y sus aplicaciones en sistemas embebidos*. Argentina: Congreso argentino de sistemas embebidos (CASE).
- [9] (2012).Terra. *NFC Forum*. Recuperado el 07 de agosto de 2014, de <http://goo.gl/wrHUxr>
- [10] Bueno M., Pavón P., De Gea A., (2011). *La tecnología NFC y sus aplicaciones en un entorno universitario*. Cartagena de Indias: Departamento de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Cartagena.

- [11] Veloz D., (2010). *Diseño e implementación de un prototipo para el control de acceso de personas aplicando la tecnología NFC por medio del uso de teléfonos celulares compatibles con esta tecnología*. Ecuador: Escuela Politécnica Nacional.
- [12] Messerschmidt T., (2014). *Guía a la galaxia de aplicaciones móviles*. [en línea] Alemania: Enough Software. Disponible en: <http://goo.gl/TRhIFg>
- [13] Janssen C., *Definition - What does Mobile Application (Mobile App) mean?* .Recuperado el 07 de agosto de 2014, de <http://goo.gl/WyJ52b>
- [14] Navarro D., (2013). *Arquitectura de una aplicación móvil*. Recuperado el 07 de agosto de 2014, de <http://goo.gl/UuCnTO>
- [15] Andorid. (2013). *NFC Basics*. Recuperado el 07 de agosto de 2014, de <http://goo.gl/65FJ5>
- [16] Aplicación Trigger. Google Play. Recuperado el 07 de agosto de 2014, de <http://goo.gl/Nb0dU>
- [17] Aplicación NFC Tools. Google Play. Recuperado el 07 de agosto de 2014, de <http://goo.gl/g3KIkM>
- [18] Aplicación NXP Writer. Google Play. Recuperado el 07 de agosto de 2014, de <http://goo.gl/4E39s>
- [19] (2014). *List of NFC-enabled mobile devices*. Recuperado el 07 de Agosto de 2014, de <http://goo.gl/UKRN3>
- [20] Cantone, D. (2006). *Implementacion y Debugging*. MP Ediciones.
- [21] Albaladejo, X. (s/a). Desarrollo iterativo e incremental. Recuperado el 07 de Agosto de 2014, de <http://goo.gl/2KNai>

[22] The Apache *Software* Foundation. (2012). Licencias (en inglés). Recuperado el 07 de Agosto de 2014, de <http://goo.gl/fcKD14>

[23] IEEE Computer Society. (1999). "Science and Engineering for Software Development: A Recognition of Harlan D. Mills' Legacy" About Harlan D. Mills.

[24] DENSO WAVE INCORPORATED. (s/a). What is a QR Code? (en inglés). Recuperado el 15 de Agosto de 2014, de <http://goo.gl/VZN4GJ>

[25] González, D. (2013). Codificación Huffman. Recuperado el 15 de Febrero de 2015, de <http://goo.gl/hTqfxJ>

Anexo A: Código QR

Un código de respuesta rápida o QR (del inglés *Quick Response*) es la marca comercial de un tipo de código de barras matricial (o código de barras bidimensional) diseñado por la industria automotriz en Japón. Un código de barras es una etiqueta óptica de lectura mecánica que contiene información sobre el objeto al que se encuentra unido. Un código QR usa cuatro modos de codificación estandarizados (numérico, alfanumérico, byte/binario y *kanji*) para almacenar datos de forma eficiente. ^[24]

Este tipo de códigos se hizo popular fuera de la industria automotriz debido a su facilidad y rapidez de lectura además de su gran capacidad de almacenamiento comparados con los códigos de barra UPC (*Universal Product Code* o Código Universal de Producto) estándar, utilizados globalmente para la identificación y rastreo de productos en tiendas y almacenes. Las aplicaciones de los códigos QR incluyen rastreo de productos, identificación, seguimiento del tiempo, administración de documentos y mercadeo en general.

Un código QR se compone de módulos negros (puntos cuadrados) dispuestos en una matriz cuadrada sobre un fondo blanco, el cual puede ser leído por un dispositivo de captura de imágenes, como una cámara fotográfica, y procesado utilizando la corrección de error Reed-Solomon hasta que la imagen pueda ser interpretada correctamente. Los datos solicitados son extraídos de los patrones presentes en las componentes horizontales y verticales de la imagen.



www.ciens.ucv.ve

Figura A-1. Código QR generado a partir del URL <http://www.ciens.ucv.ve>
Fuente: Creada por el autor.

Anexo B: Codificación Huffman

La codificación Huffman es un algoritmo usado para compresión de datos. El término se refiere al uso de una tabla de códigos de longitud variable para codificar un determinado símbolo (como un carácter), donde la tabla ha sido rellena de una manera específica basándose en la probabilidad estimada de aparición de cada posible valor de dicho símbolo.^[25]

El algoritmo consiste en la creación de un árbol binario que tiene cada uno de los símbolos en una hoja, y construido de tal forma que siguiéndolo desde la raíz a cada una de sus hojas se obtiene el código Huffman asociado.

Algoritmo de codificación Huffman

1. Crear un nodo hoja para cada símbolo, asociando un peso según su frecuencia de aparición e insertarlo en la lista ordenada ascendentemente.
2. Mientras haya más de un nodo en la lista:
 - a. Eliminar los dos nodos con menor probabilidad de la lista.
 - b. Crear un nuevo nodo interno que enlace a los nodos anteriores, asignándole como peso la suma de los pesos de los nodos hijos.
 - c. Insertar el nuevo nodo en la lista, (en el lugar que le corresponda según el peso).
3. El nodo que quede es el nodo raíz del árbol.

Con este árbol generado se puede conocer el código asociado a un símbolo, así como obtener el símbolo asociado a un determinado código.

Para obtener el código asociado a un símbolo se debe proceder del siguiente modo:

1. Comenzar con un código vacío.
2. Iniciar el recorrido del árbol en la hoja asociada al símbolo.
3. Comenzar un recorrido del árbol hacia arriba.
4. Cada vez que se suba un nivel, añadir al código la etiqueta de la rama que se ha recorrido.
5. Tras llegar a la raíz, invertir el código.
6. El resultado es el código Huffman deseado.

Un ejemplo de un árbol dibujado se observa en la figura B-1 donde los nodos hijos finales son a los que les corresponde un símbolo y una frecuencia, y los nodos padres de estos últimos se conectan como pequeños árboles a otros nodos. Los nodos con puros números indican la suma de las frecuencias de los nodos hijos correspondientes.

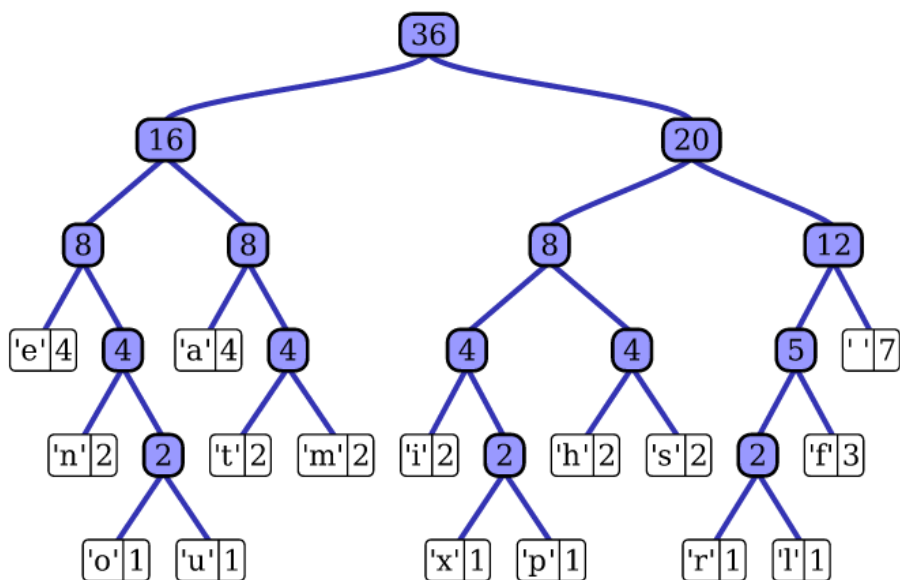


Figura B-1. Árbol Binario generado a partir de la frecuencia exacta de los caracteres en la frase "this is an example of a huffman tree"

Fuente: González,R. (2013). Codificación Hoffman.

Anexo C: Encuesta utilizada en la prueba de aceptación de la aplicación

Prueba de aceptación aplicación tagR

***Obligatorio**

Tras una primera vista, ¿le queda claro cuál es el objetivo de la aplicación ? *

☐ Si

☐ No

¿Que tan intuitiva encuentra la interfaz? *

☐ Poco intuitiva

☐ Intuitiva

☐ Muy intuitiva

¿Cómo calificaría la navegabilidad de la aplicación ? *

☐ Mala

☐ Regular

☐ Buena

☐ Muy buena

☐ Excelente

¿Cómo calificaría los colores utilizados en la aplicación ? *

☐ Adecuados

☐ No adecuados

¿Cómo calificaría la cantidad promedio de elementos presentes en la interfaz ? *

☐ Pocos

☐ Suficientes

☐ Demasiados

¿Cómo calificaría el tiempo promedio de carga de los elementos de la interfaz ? *

☐ Muy lento

☐ Lento

☐ Normal

☐ Rápido

☐ Muy rápido

Observaciones

Escriba cualquier opinión que desee agregar y no haya sido respondida en alguna de las preguntas anteriores

Figura C-1. Fragmento de la encuesta utilizada durante las pruebas de aceptación.
Fuente: Creada por el autor.

Resultados de la encuesta

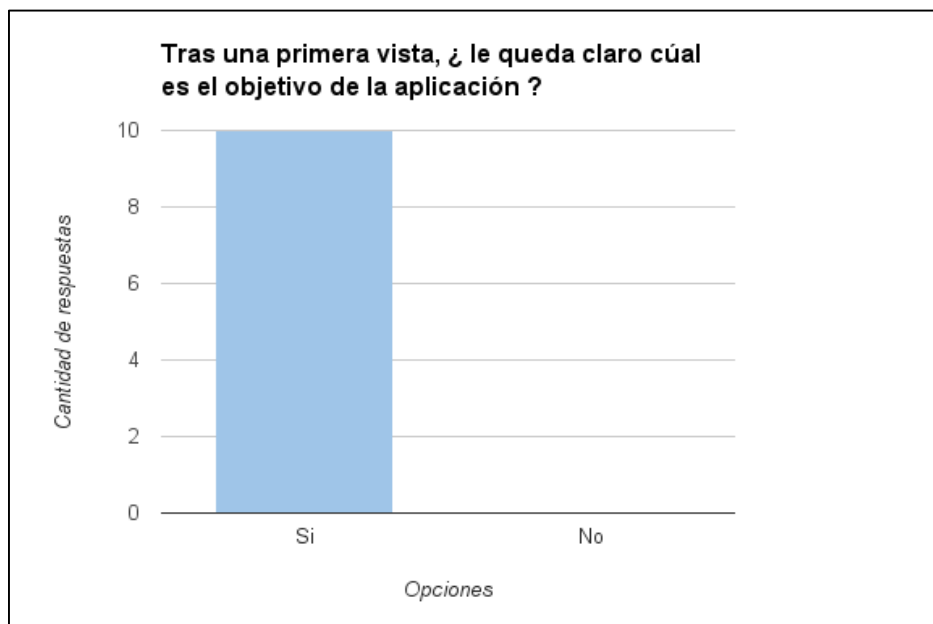


Figura C-1. Gráfica de cuantificación de respuestas de la pregunta n° 1.
Fuente: Creada por el autor.

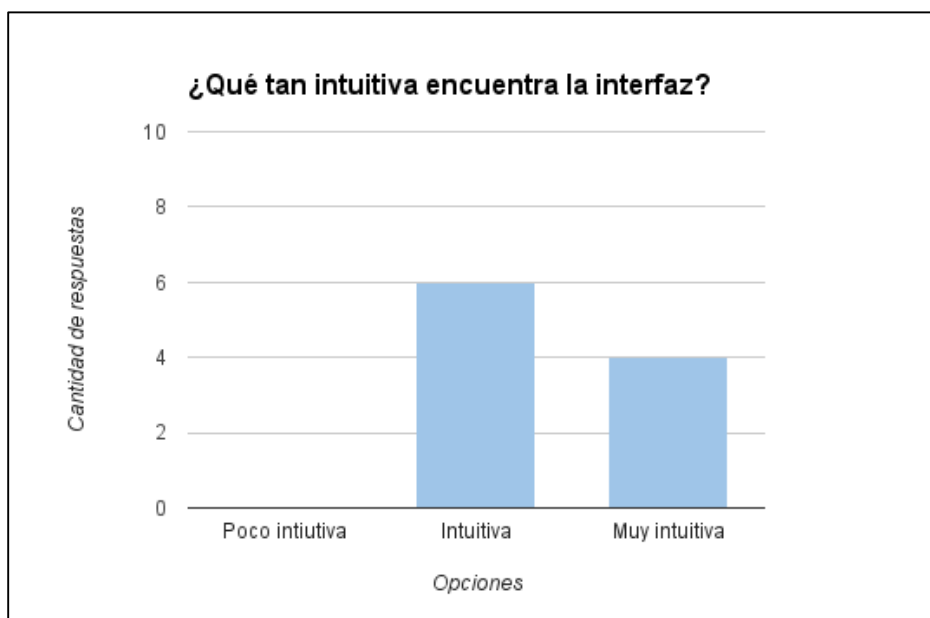


Figura C-2. Gráfica de cuantificación de respuestas de la pregunta n° 2.
Fuente: Creada por el autor.

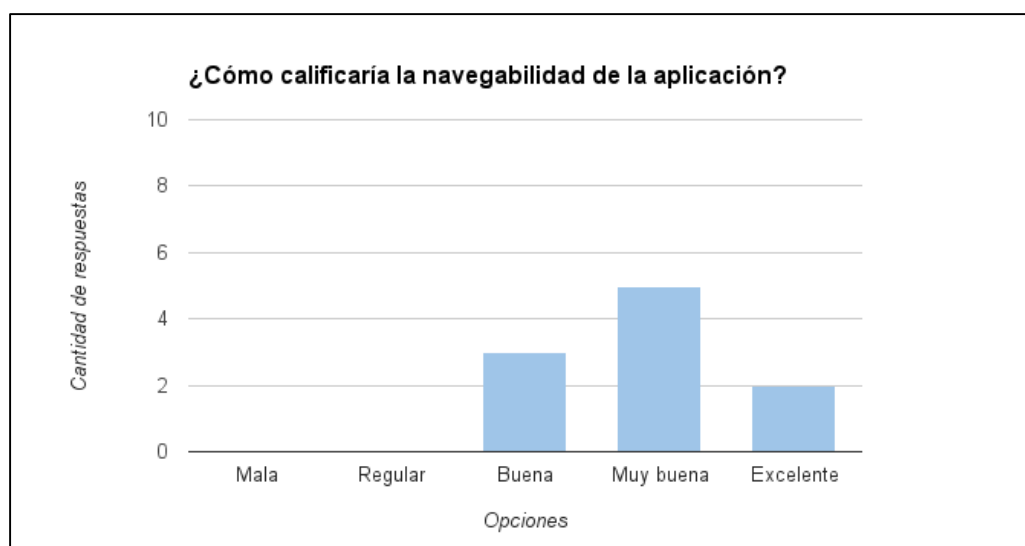


Figura C-3. Gráfica de cuantificación de respuestas de la pregunta n° 3.
Fuente: Creada por el autor.

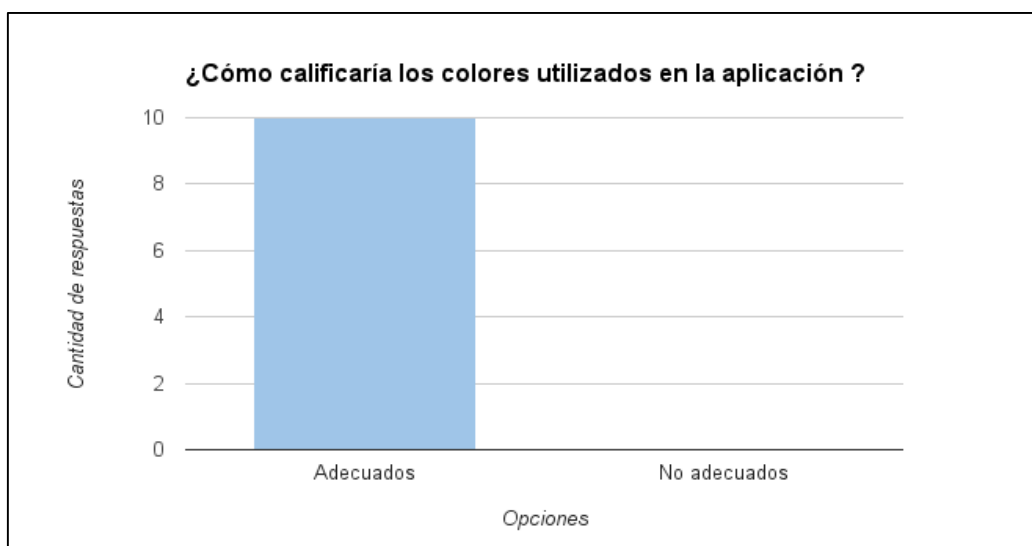


Figura C-4. Gráfica de cuantificación de respuestas de la pregunta n° 4.
Fuente: Creada por el autor.

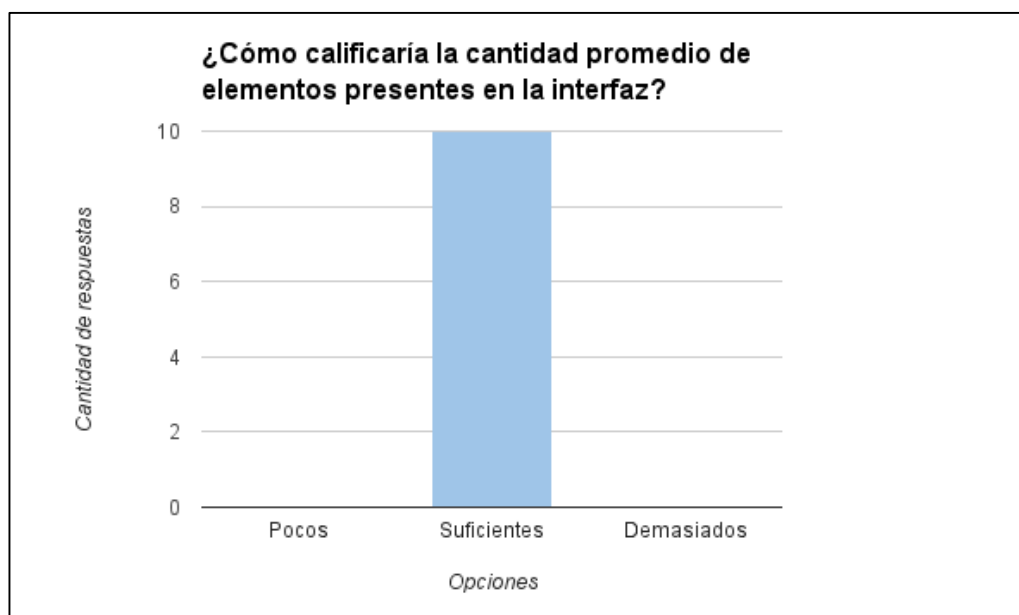


Figura C-5. Gráfica de cuantificación de respuestas de la pregunta n° 5.

Fuente: Creada por el autor.

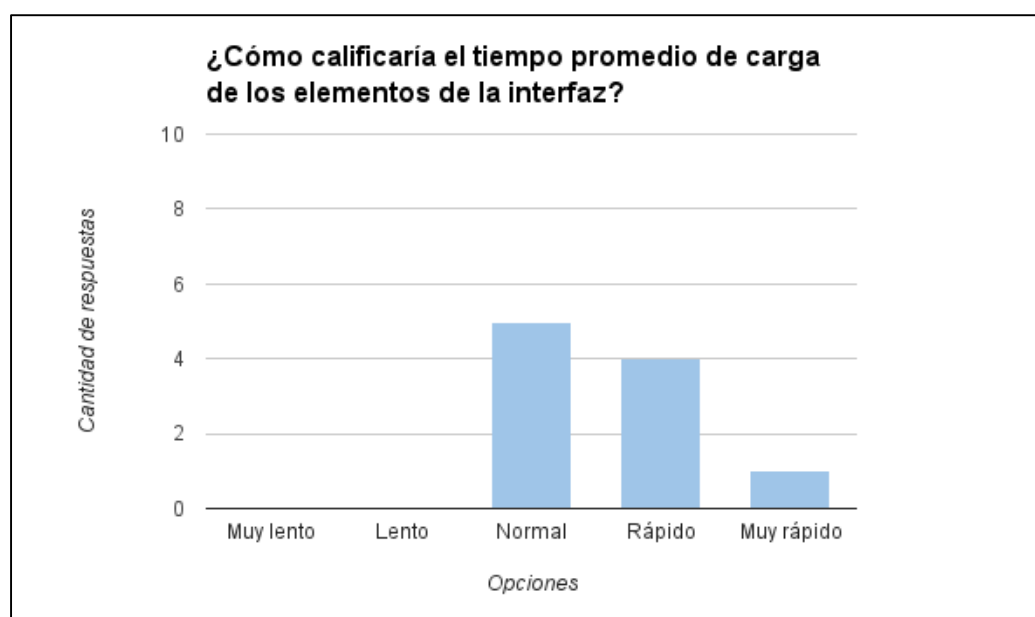


Figura C-6. Gráfica de cuantificación de respuestas de la pregunta n° 6.

Fuente: Creada por el autor.