

# Gestión de correspondencia física a través de aplicación móvil y códigos QR

Yuri V. Nieto A.<sup>1</sup>, José F. López Q.<sup>2</sup>, Carlos E. Montenegro M.<sup>3</sup>

Yuri\_nieto@cun.edu.co, cemontenegrom@udistrital.edu.co, jflopezq@hotmail.com

<sup>1</sup> Facultad de Ingeniería, CUN, Bogotá, Colombia

<sup>2</sup> Facultad de Ingeniería, Universidad Distrital FJC, Bogotá, Colombia

<sup>3</sup> Facultad de Ingeniería, Bogotá, Colombia

**Pages:** 201–213

**Resumen:** Este artículo presenta la explicación y la descripción de la arquitectura interna del desarrollo de una aplicación de elementos web y móvil para la gestión de la correspondencia física a través de códigos QR, además de las herramientas utilizadas para desarrollo y modelado para satisfacer las necesidades del usuario, que se mostrará como un requisito y entonces los casos de uso serán considerados, el proyecto se ejecutará a través de una metodología que se compone de dos partes, la primera de la investigación y la segunda del desarrollo complementan entre sí para finalmente obtener una aplicación que le da la notificación oportuna de la información contenida en la correspondencia física.

**Palabras claves:** Códigos QR, aplicaciones móviles, aplicación web, correspondencia física, desarrollo, ciclo de vida, modelado.

## *Physical correspondence management through mobile application and QR codes*

**Abstract:** This article presents the explanation and description of the internal architecture of the development of an application with web elements and mobile elements for the management of physical correspondence through QR codes, in addition to the tools used for development and modeling To satisfy the needs of the user, which will be shown as a requirement and then the use cases will be considered, the project will be executed through a methodology that is composed of two parts, the first of research and the second of development that complement each other to finally obtain an application that gives timely notification of the information that is contained in the physical correspondence.

**Keywords:** QR codes, mobile application, web application, physical correspondence, development, lifecycle, modeling.

## 1. Introducción

En la actualidad es necesario dar solución a ciertos problemas comunes que se dan a diario pero no se les presta mucha atención, en este caso, y con ayuda de la tecnología se realizará la implementación y desarrollo de una aplicación para gestionar la correspondencia física y evitar que la información de esta se pierda o que no se vea a tiempo.

El artículo está compuesto por una metodología en donde se realizará y mostrará la forma en que se obtuvieron los datos y como se realizó el análisis de estos para obtener las necesidades del usuario, también se mostrará la forma en que fue desarrollado el proyecto para cumplir con los requerimientos planteados por los desarrolladores y aprobados por el cliente, luego se realizará un marco teórico donde se hablará de la historia de los códigos QR y de los sustentos del por qué estos son tan importantes para la sociedad actual. Para complementar esta información, se mostrarán algunos ejemplos de estos códigos y como última parte del marco teórico se compararán las aplicaciones web con las aplicaciones móviles para determinar el tipo de aplicación necesario para el proyecto.

Adicionalmente, se presenta una sección de desarrollo e implementación, en la que primero se muestra la estructura de la aplicación donde se evidencia el ambiente de desarrollo utilizado, el sistema operativo para realizar la ejecución de la aplicación y el funcionamiento de la base de datos, luego se mostrarán y se hará una descripción de las tecnologías a utilizar para el desarrollo y en la última parte dentro de la implementación y el desarrollo se mostrará el modelamiento del sistema comenzando por los requerimientos y sus consecuentes casos de uso, esta parte del artículo será gráfica, donde se mostrarán ciertos diagramas y su correspondiente descripción y explicación, el primero de estos diagramas es el diagrama de clases para mostrar la estructura de la lógica de lo desarrollado, seguido del diagrama de secuencia para mirar los procesos a lo largo del tiempo, el diagrama de colaboración para ver la interacción entre los roles y el diagrama de componentes para mostrar los distintos elementos que son necesarios para el funcionamiento de la aplicación.

## 2. Metodología

El proceso de investigación aplicado para este proyecto se dividió en dos partes, la primera de ellas, el uso de un enfoque de investigación y por otro lado el uso de una metodología de desarrollo.

Como metodología de investigación se decidió el uso del enfoque cuantitativo, esta decisión se tomó debido a la forma en que se quiere adquirir la información. Para este caso, es necesario la obtención y el análisis de datos a nivel de encuestas, por lo que el enfoque cuantitativo brinda suficientes bondades para que al asignar números a estos resultados se puedan generalizar resultados a nivel poblacional.

Para llevar a cabo de forma correcta el análisis de datos a través del enfoque cuantitativo es necesario cumplir con ciertos pasos, primero se debe plantear y delimitar el problema, donde se especifique una pregunta a resolver, luego se debe realizar la revisión de la literatura, es decir, la revisión de trabajos o estudios anteriores relacionados con el tema que se está investigando, como siguiente paso, se construye el marco teórico para poder

realizar la hipótesis de la investigación, la cual a partir del marco teórico y de la revisión de la literatura se pueden consolidar consistentemente, o por el contrario, descartarlas por falta de evidencia.

Con ayuda del procedimiento mencionado anteriormente, se escoge el enfoque cuantitativo para apoyar la investigación, de tal forma, que los resultados obtenidos arrojen características que se puedan tener en cuenta para el correcto desarrollo del proyecto.

Por otro lado, se tiene la metodología de desarrollo, en la cual se escogió el ciclo de vida iterativo (Campderrich Falgueras, 2013) por las bondades que brinda. Este ciclo de vida está basado en el modelo de cascada, (Maller, Ochoa, & Silva, 2004), (Hossein & Natsu, 1997) el cual se separa en cuatro fases, comenzando por un análisis, seguido de un diseño, se hace la correspondiente codificación y se sigue para realizar las pruebas. Luego de completadas cada una de las 4 fases, se obtiene un resultado también llamado liberable que será entregado al cliente para su correspondiente aprobación y realización de sugerencias para empezar con el siguiente ciclo; este proceso se realiza cuantas veces sea necesario hasta cumplir con los requerimientos acordados al principio del proyecto. (Carnegie Mellon, 2008)

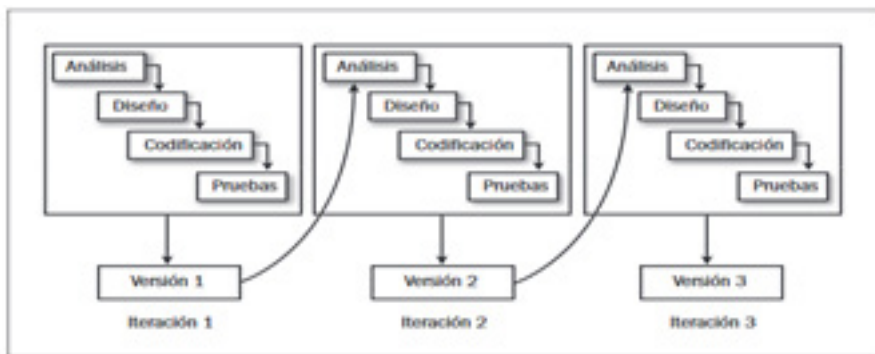


Figura 1 – Ciclo de Vida

En la figura anterior, se puede ver de forma gráfica el funcionamiento del ciclo de vida iterativo, para el caso de la imagen se realizan tres iteraciones y por lo tanto se obtienen tres entregables que son las versiones y se debe tener en cuenta que la cantidad de versiones (liberables) debe corresponder a la misma cantidad de iteraciones.

### 3. Marco teórico

#### 3.1. Historia

A lo largo de la historia de la humanidad la tecnología ha jugado un papel muy importante para la optimización de tiempos y recursos en ciertas tareas, en la década de 1960, para realizar el registro de productos para la venta era necesario el registro y búsqueda manual de los productos, esta tarea llevaba mucho tiempo y el uso de mucho más personal que se veía afectado debido al alto nivel de trabajo.

Por esta razón, en el año 1952, se inventaron los códigos de barras y fueron usados comercialmente hasta 1966 (Meneses Fernández, Martín Gutiérrez, & Álvarez Martínez, 2014), los cuales son una serie de líneas de distinto grosor que almacenan cierta información, con esta invención se realizó el desarrollo del sistema POS en la década de 1970 (Gutiérrez, 2012), en dónde se almacenaba este código de barras se adhería al producto, en el código estaba almacenado la información correspondiente al precio y al realizar la lectura esta información se obtenía instantáneamente.

Denso Wave, en 1994, anunció el lanzamiento de un nuevo código con muchas más bondades que las brindadas por el código de barras, este nuevo código era capaz de almacenar 7.000 caracteres y podía ser leído 10 veces más rápido, por esto mismo se le denominó código de Respuesta Rápida, del inglés Quick Response, al cual llamaron simplemente código QR. (Leiva-Aguilera, 2012)

Luego de esto, el código QR se empezó a utilizar como estándar para muchas empresas y asociaciones, la primera de ellas fue la AIM (Automatic Identification Manufacturer), para la identificación automática de la fabricación a nivel industrial, luego de ello, se estandarizó en otras empresas japonesas, para que finalmente en el año 2000, fuera aprobado por la ISO (International Organization for Standardization) encargada de regir los estándares de la industria a nivel mundial.

Otro detalle que ayudó a la difusión del código QR fue la decisión de Denso Wave de liberar el código fuente para el uso de este nuevo método de almacenamiento de información, por esta misma razón en 2002, los teléfonos móviles se empezaron a comercializar con la opción de leer códigos QR (Villoldo, Salom, Pons Chaigneau, Rubio Montero, & Vallés Navarro, 2012) que permitían el acceso a páginas web, de esta forma la cantidad de información que se podía obtener sobre algún producto o servicio era mucho mayor.

### 3.2. Ejemplos códigos QR

Actualmente existe una gran variedad de códigos QR a lo largo del mundo (Luque Ordóñez, s.f.), los ejemplos que se mostrarán a continuación son estándares de alguna empresa o asociación que a través de estos han optimizado o mejorado alguna característica dentro de ellas.



Figura 2 – Aztec Code

El primero de ellos es el Aztec Code, apareció en el año de 1995, inventado por inventado por Andrew Longacre, Jr. y Robert Hussey, su nombre se debe a la similitud con las pirámides aztecas, tiene la ventaja de mejorar el uso del espacio porque no necesita de una zona para determinar el sentido del código.



Figura 3 – Micro QR Code

El siguiente de ellos, es el código QR micro, que es una representación minimizada del código QR estándar, a pesar de que minimiza el uso del espacio la cantidad de caracteres que puede almacenar también se ve reducida.



Figura 4 –Datamatrix

Como bien lo dice su nombre, este código QR, se asemeja a una matriz de datos, este tipo de código puede almacenar muchas más caracteres que otros, es capaz de almacenar hasta 2.335 caracteres alfanuméricos, es mayormente usado en los dispositivos electrónicos de menos tamaño como lo recomienda la Alianza de Industrias Electrónicas (EIA – Electronic Industries Alliance).



Figura 5 – Microsoft Tag

El último de los ejemplos, es el Microsoft Tag, que como se evidencia en su nombre, pertenece a la multinacional Microsoft. Este fue un servicio creado en el año 2010, con el objetivo de que cualquier persona promoviera algún servicio en la web por medio de este y que con una aplicación del mismo proveedor se pudiera interpretar, sin embargo, en 2013, Microsoft cerró este proyecto debido a la falta de popularidad de este tipo de código.

### **3.3. Aplicación Web vs. Aplicación Móvil**

Para determinar el tipo de aplicación sobre el que se iba a desarrollar, fue pertinente realizar una comparación entre las ventajas que brindan las aplicaciones web y las

aplicaciones móviles, para realizar esta comparación se mostrarán las características de cada una de ellas y luego se define la que brinda la mayor cantidad de ventajas para este proyecto.

En primer lugar, se tienen las aplicaciones web las cuales se ejecutan desde un navegador, tienen las ventajas de ser utilizadas sin necesidad de una instalación previa, dentro de las ventajas que se tienen dentro de este tipo de aplicaciones está la posibilidad de almacenar cierta información en servidores que pueden ser accedidos cuando se tiene internet, con una conexión de este tipo y un navegador la información puede ser consultada desde cualquier parte del mundo.

Algunos ejemplos de aplicaciones web que se usan a diario están las aplicaciones de correo electrónico de las distintas empresas proveedoras de este servicio en el mundo como lo son Google, Yahoo o Microsoft; otros ejemplos, son la variedad de redes sociales que hay actualmente dentro de las cuales están Facebook, Twitter, entre otras.

Con información general de que son las aplicaciones web y algunos ejemplos de estas, es pertinente hablar de las ventajas, este tipo de aplicaciones tienen una compatibilidad multiplataforma, es decir, que sin importar el sistema operativo en el que se encuentre o el dispositivo se podrá acceder a la aplicación.

Por otro lado, se tienen las aplicaciones móviles que hoy en día, son importantes para el desempeño de la cotidianidad de las personas, debido a la gran cantidad de aplicaciones que existen en el mercado se pueden realizar diferentes actividades y esto es lo que ha aumentado la popularidad de estas, las principales categorías con las que se interactúan son juegos, educación, negocios, estilo de vida y entretenimiento.

El uso de aplicaciones móviles o “apps” está desplazando el uso de aplicaciones web dado que en ocasiones es necesario realizar acciones fuera de línea y a veces no se tiene acceso a internet desde el dispositivo móvil lo que permite que las ventajas de este tipo de aplicaciones aumenten.

Al analizar las distintas ventajas que brinda cada uno de los tipos de aplicaciones se puede concluir que cada una es pertinente en su contexto, es decir, que dependiendo de la necesidad del usuario se utilizará una sobre la otra, para este caso se realizará uso de los dos tipos de aplicación para satisfacer los requerimientos del proyecto.

## **4. Desarrollo e implementación**

### **4.1. Estructura de la aplicación**

1. *Ambientes de desarrollo:* Para desarrollar esta aplicación es necesario de dos ambientes, el primero de ellos, Java para el desarrollo de la parte móvil de la aplicación y HTML5 (Xuelei, Shulin, Guyang, & Liping, 2013) con uso de Javascript (Elliott, 2014) para la parte web de la aplicación.

Comenzando por la parte móvil, se usa Java por el soporte multiplataforma que brinda, porque este es un lenguaje de programación orientado a objetos, tiene una gran oferta de librerías para el desarrollo de distintas aplicaciones incluidas algunas para complementar el desarrollo para aplicaciones Android. Además, es un lenguaje fácil de aprender, de

leer y de entender, y adicionalmente con ayuda de las plataformas de desarrollo (IDE) se facilita el trabajo y se reducen la cantidad de errores que se puedan cometer.

Por otra parte, se tiene el desarrollo web con HTML5 (Harris, 2014) y Javascript, con estas herramientas se puede realizar un despliegue liviano de información hacia cualquier dispositivo, además de ello los diseños se pueden optimizar de tal forma que la visualización sea buena en cualquier tamaño de pantalla, a pesar de las ventajas el desarrollo web para ser ejecutado desde dispositivos móviles presenta algunas dificultades como el acceso al hardware como lo es la cámara.

2. *Sistema operativo:* Se realizará uso del sistema operativo Android el cual está basado en Linux, en este sistema operativo se puede realizar la ejecución de aplicaciones móviles sin ningún problema. A pesar de que este sistema operativo brinda grandes ventajas en tema de seguridad para la instalación y ejecución de aplicaciones por parte de fuentes desconocidas, también se tiene la opción de omitir esta parte del sistema de seguridad con el objetivo de ejecutar aplicaciones propias.

3. *Base de datos:* El almacenamiento de datos se debe realizar de una forma en la que estos estén relacionados y agrupados de una forma estructurada, para esto, es necesario el uso de base de datos, además de ello la seguridad de la información aumenta junto con la cantidad de funcionalidades que se pueden hacer sobre esta.

Existen varios modelos para realizar una base de datos, para este caso, se utilizará el modelo relacional. Para este tipo de modelo, es necesario realizar el modelamiento para que cumpla con las especificaciones del cliente.

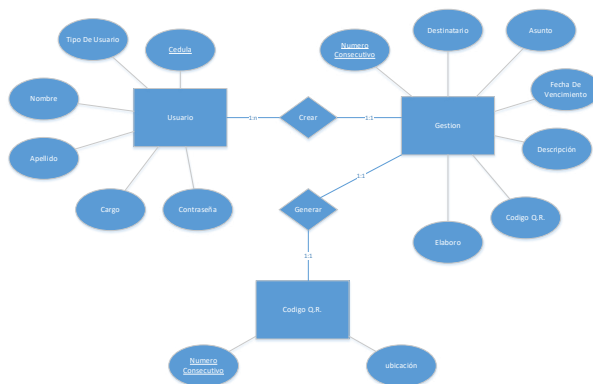


Figura 6 – Diseño físico de la base de datos

El anterior diagrama entidad relación, es el obtenido para el desarrollo y diseño físico de la base de datos donde se tienen tres entidades: usuario, gestión y código QR, que a su vez, tienen cierta cantidad de atributos para guardar la información, cabe resaltar que no todas las entidades se relacionan, usuario se relaciona con gestión y gestión con código QR, pero las entidades usuario y código QR nunca se relacionan, esto se hace para evitar la alteración de datos por parte del usuario.

## 4.2. Tecnologías a utilizar

1. *Android Studio*: (Van Drongelen, 2015) Es un entorno de desarrollo integrado (IDE) para desarrollar aplicaciones móviles del sistema operativo Android, fue el reemplazo de Eclipse y su primera versión estable fue publicada en diciembre de 2014, tiene características que permiten la optimización de recursos, entre las cuales se encuentran la renderización en tiempo real, plantillas para la creación de diseños comunes además de la posibilidad de crear estos con la opción de agarrar y soltar, y la ejecución virtualizada de la aplicación para realizar las pruebas pertinentes a las aplicaciones.
2. *MySQL (PHPMyAdmin)*: (Rochkind, 2013) Con esta herramienta desarrollada en PHP se realiza la administración de la base de datos en MySQL a través de una página web, dentro de esta aplicación se pueden importar y exportar datos desde y hacia diferentes formatos, creación de consultas complejas utilizando Query-by-Example (QBE) y la posibilidad de monitoria las actividades y recursos asignados a la base de datos.
3. *Visio 2013*: Es una aplicación de Microsoft Office para realizar diagramas y gráficas, con ayuda de esta herramienta se buscan realizar los distintos diagramas como lo son los diagramas de caso de uso, el modelo entidad relación para la base de datos y los diagramas de arquitectura: diagrama de clases, diagrama de secuencia, diagrama de colaboración y diagrama de componentes.
4. *Dreamweaver*: Dreamweaver es un programa de estudio de Adobe para la construcción, diseño y edición de sitios y aplicaciones web, actualmente con ayuda de Dreamweaver es posible la realización de páginas sin necesidad de conocer HTML, además soporta herramientas que complementan la funcionalidad como los son las hojas de estilo (CSS) y la realización de lógica con Javascript, esta herramienta está disponible en Windows y en MAC, aunque también es posible utilizarlos en UNIX con simulador de herramientas de Windows como Wine.
5. *Sublime Text*: Es un editor de texto y editor de código que soporta una gran cantidad de idiomas y tiene varias bondades dentro de las que se encuentran la multiplicidad de la selección, la multiplicidad del cursor, búsqueda dinámica, auto completado y marcado de llaves, uso de pestañas para la edición de varios archivos al mismo tiempo, resaltado de paréntesis para la identificación de bloques.
6. *Selenium*: Con esta herramienta se tiene un entorno de pruebas de software principalmente para aplicaciones basadas en la web, estas pruebas se pueden realizar sin necesidad de escribirlas en un lenguaje de scripting, además de ellos se pueden realizar depuración y puntos de verificación (breakpoints).
7. *JMeter*: (Erinle, 2014) Esta última herramienta desarrollado por Apache brinda la posibilidad de hacer pruebas de desempeño en tiempo real de servicios especializados en aplicaciones web, además de ello tiene la opción de monitorear los recursos que se están utilizando, la primera versión de JMeter fue lanzada el 9 de marzo de 2001.



### 4.3. Modelamiento del sistema

1. *Requerimientos*: Para realizar los requerimientos fue necesario realizar una encuesta para conocer las necesidades que tiene la empresa respecto a la correspondencia física, al realizar un análisis de los datos y estadísticas obtenidas por medio del enfoque cuantitativo se levantaron cuatro requerimientos.

El primero de ellos, con identificador RF-001, llamado “Acceso a la aplicación”, este tiene una prioridad alta y consta en que la aplicación será accesible por medio de un usuario y contraseña previamente configurados en el sistema. En segundo lugar, está el requisito de “Administración de usuarios”, con identificador RF-002 y con prioridad alta, en donde la aplicación estará en la capacidad de realizar la creación, modificación y eliminación de usuarios; el tercer requerimiento, “Documentos”, identificado con RF-003 y con alta prioridad, también es en donde la aplicación estará en la capacidad de realizar la creación, modificación y eliminación de documentos que contengan la información relevante de la correspondencia recibida, y el último requerimiento, “Notificaciones”, con identificador RF-004 y alta prioridad, de las misma forma que los otros requerimientos, tiene como descripción que la aplicación estará en la capacidad de realizar la creación, modificación y eliminación de las alarmas que servirán.

2. *Casos de Uso*: A partir de los requerimientos levantados anteriormente, se procede a realizar la elaboración de los casos de uso, de los cuales se obtienen 5, los cuales serán realizados por 3 actores y su resultado se evidencia en el siguiente diagrama, en el que se aprecia quien tiene asignado el caso de uso y las precondiciones para poder realizarlo.

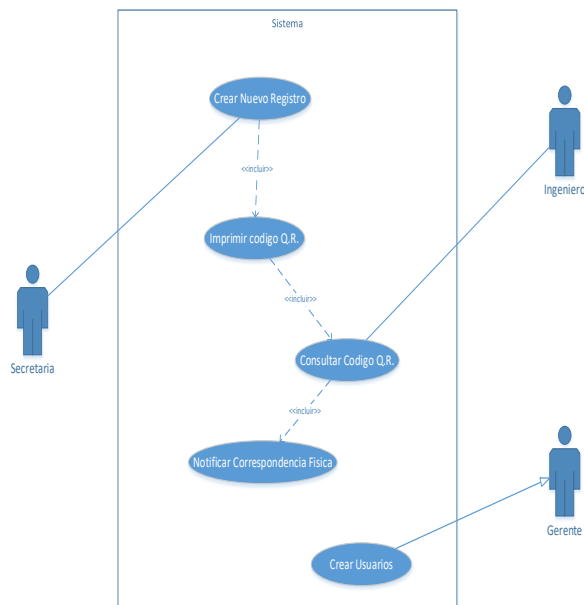


Figura 7 – Casos de uso

3. *Diagrama de clases:* En el siguiente diagrama de clases, se representa la estructura de la lógica realizada en Java a través del Android Studio, es relevante recordar que los diagramas de clases según UML (Unified Model Language) están compuestas por tres secciones horizontales, la primera de ellas lleva el nombre de la clase, la segunda los atributos y la tercera los métodos dentro de la clase.

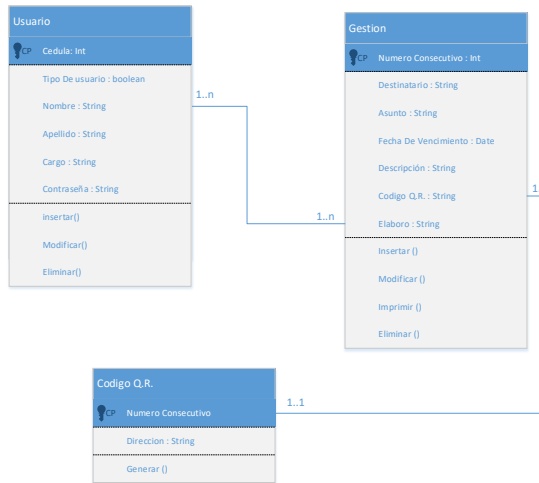


Figura 8 – Diagrama de Clases

En este diagrama, se puede ver que el desarrollo de la aplicación móvil se realizó en tres (3) clases, la primera de ellas “usuario”, compuesta por 5 atributos y 3 métodos que correspondiendo con lo planteado en los requerimientos son insertar, modificar y eliminar; la segunda clase es “Gestión”, en donde fuera de insertar, modificar y eliminar, tiene la opción de imprimir, y finalmente, está la clase código “QR” en donde únicamente se puede realizar la generación del código y solo tiene como atributo la dirección en donde está almacenada la imagen.

4. *Diagrama de secuencia:* Para visualizar la interacción de los objetos a través del tiempo se tienen los diagramas de secuencia, en el diagrama a continuación, se evidencia la existencia de 5 objetos: Secretaria, Formulario, Generador Códigos QR, Destinatario y Android, en la primera parte del diagrama se ve el envío de 5 mensajes de Secretaria a Formulario, estos mensajes forman la parte de un nuevo registro, luego de ello, se realiza la parte de la generación del código y la verificación del mismo, para que finalmente, como se ve en los últimos 2 mensajes enviados, entre Destinatario y Android para la notificación para no dejar pasar las fechas en la correspondencia física.

5. *Diagrama de colaboración:* En un diagrama de colaboración se ve evidenciada la interacción entre los distintos roles, para el diagrama a continuación, hay dos roles “Secretaria” y “Destinatario”, en donde la Secretaria se encarga de realizar el ingreso de datos a través de la plataforma de la aplicación web que está realizada en PHP y esto se realiza para la generación del código QR el cual es enviado a la impresora para que

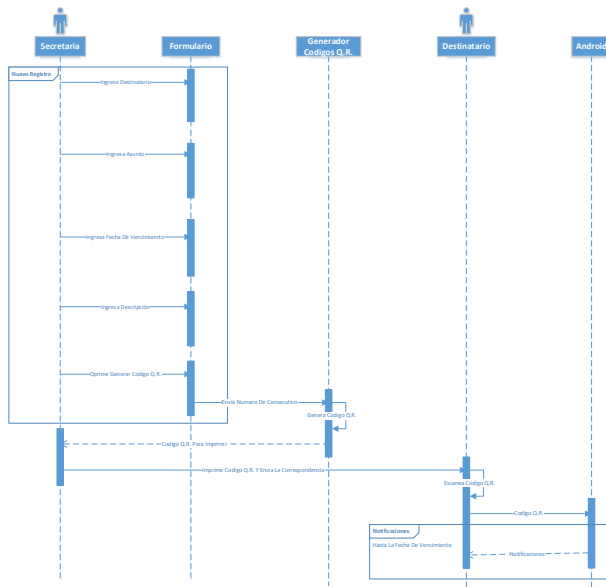


Figura 9 – Diagrama de secuencia

este sea anexo a la correspondencia que será enviada al Destinatario, para que este con ayuda de la aplicación móvil escanee el código QR que será interpretado por la misma aplicación y muestre la información para la correspondiente notificación.

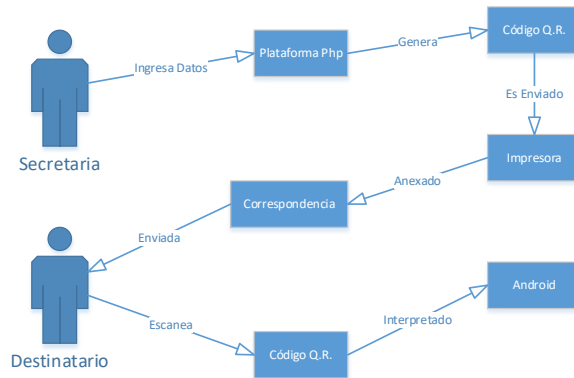


Figura 10 – Diagrama de Colaboración

6. *Diagrama de componentes:* En el siguiente diagrama de componentes, se muestra la interacción del actor con los distintos elementos que componen el sistema, a nivel general hay tres (3) sistemas: Estación de trabajo, Sistema y Android, dentro de cada uno de ellos hay componentes que se relacionan con los componentes de los otros

sistemas. Para el sistema Estación de trabajo, compuesto por Explorador web se tienen que es accedido por parte de los otros dos componentes, Sistema y Andorid, dentro de cada una de ellos existe su correspondiente IU (Interfaz de Usuario) que por otra parte están trabajando con el componente SGDB (Sistema de Gestión de Base de Datos) para adicionar, modificar y eliminar los datos.

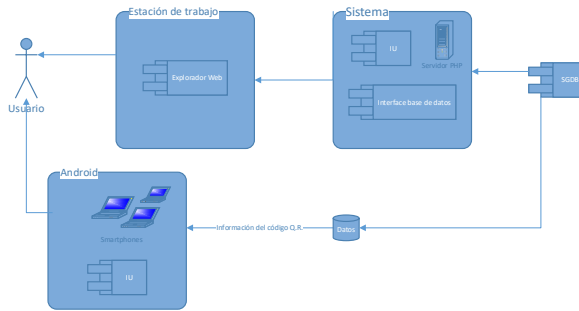


Figura 11 – Diagrama de Componentes

## 5. Conclusiones

A través de la metodología con su enfoque cuantitativo se mencionaron las distintas formas en las que se determinaron las necesidades del usuario para darles una prioridad, esto se hizo al plantear los requerimientos y consecuentemente los casos de uso, complementando la metodología se usó el ciclo de vida iterativo compuesto por varios ciclos de vida en cascada con lo que se obtuvieron ciertas versiones de la aplicación las cuales contenían mejoras y adelantos a medida que los ciclos iban pasando finalmente se obtuvo una aplicación con sus componentes web y móvil que satisface los requerimientos planteados.

Por otro lado, se realizó una contextualización correcta a través del marco teórico para que en la sección de desarrollo e implementación se entendiera la estructura de la aplicación, las tecnologías utilizadas, y finalmente, el modelamiento del sistema que mostró los distintos diagramas para entender la arquitectura interna de la aplicación desde distintas perspectivas incluyendo la parte de desarrollo con los diagramas de clases y componentes, y la interacción de los roles con las tareas a realizar a lo largo del tiempo con los diagramas de secuencia y colaboración.

## Referencias bibliográficas

- Campderrich Falgueras, B. (2013). *Ingeniería del software*. Editorial UOC.
- Carnegie Mellon, U. (2008). *CMMI Destilled* (Tercera Edición ed.). Estados Unidos de America: Addison-Wesley.
- Elliott, E. (2014). *Programming JavaScript Applications: Robust Web Architecture with Node, HTML5, and Modern JS Libraries*. Sebastopol, CA : O'Reilly Media.

- Erinle, B. (2014). *JMeter Cookbook*. Birmingham: Packt Publishing.
- Gutiérrez, F. (2012). CODIGO QR EN BIBLIOTECAS Y ALFABETIZACION INFORMACIONAL MOVIL. *Revista Chilena de Bibliotecología*.
- Harris, A. (2014). *HTML5 and CSS3 All-in-One For Dummies*. Hoboken.
- Hossein, S., & Natsu, C. (1997). Characterizing a Software Process Maturity Model for Small Organizations. University of Nebraska at Omaha.
- Leiva-Aguilera, J. (2012). Introducción y algunos usos de los códigos QR. *Anuario Think EPI*, 309–312.
- Luque Ordóñez, J. (s.f.). *Códigos QR. AUTORES CIENTÍFICO-TÉCNICOS Y ACADÉMICOS*.
- Maller, J., Ochoa, J., & Silva, C. (2004). Lightening the software production process in a CMM level 5 framework. *JISBD*.
- Meneses Fernández, M., Martín Gutiérrez, J., & Álvarez Martínez, E. (2014). Uses of qr code in printed press. The audio visualization of paper. *Innovar*, 67–80.
- Rochkind, M. J. (2013). *Expert PHP and MySQL: Application Design and Development*. [Berkeley, Calif.]: Apress.
- Van Drongelen, M. (2015). *Android Studio Cookbook*. Birmingham, UK : Packt Publishing.
- Villoldo, M., Salom, A., Pons Chaigneau, D., Rubio Montero, F., & Vallés Navarro, R. (2012). *Tecnologías móviles en bibliotecas*. Valencia: Biblioteca I Documentació Científica.
- Xuelei, R., Shulin, Y., Guyang, W., & Liping, R. (2013). Research on Cross Platform Digital Publishing Technology Based on HTML5. *8th International Conference on Information Technology in Medicine and Education (ITME)*. Fuzhou.

Reproduced with permission of copyright owner. Further reproduction prohibited without permission.