

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

**REALIDAD AUMENTADA PARA DISPOSITIVOS
MÓVILES ANDROID APLICADA A VISUALIZAR
INFORMACIÓN MÉDICA**

Mario Jiménez Mancebo
Tutor: Alejandro Bellogín Kouki
Ponente: Alejandro Sierra Urrecho

JUNIO 2016

**REALIDAD AUMENTADA PARA DISPOSITIVOS
MÓVILES ANDROID APLICADA A VISUALIZAR
INFORMACIÓN MÉDICA**

**AUTOR: Mario Jiménez Mancebo
TUTOR: Alejandro Bellogín Kouki**

**Dpto. Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio de 2016**

Resumen

Hoy en día, hay muchas personas con una o varias enfermedades que requieren seguir unos tratamientos con muchos medicamentos y muchas tomas al día, y recordar cuándo deben hacer las tomas puede llegar a ser algo complicado. Este problema puede incluso ser peor para personas mayores, ya que muchas de ellas tienen problemas de memoria.

Para ayudar a estas personas con sus tratamientos, se va a hacer uso de los *smartphones* que hoy en día tienen un alto protagonismo en nuestra vida cotidiana y han llegado a estas personas mayores que antes no los usaban. Por lo tanto, se desarrollará una aplicación para dispositivos móviles Android cuyo foco principal es ayudar a estas personas, por ello se buscará una interfaz clara y sencilla para facilitar el manejo a estas personas mayores que no están tan familiarizadas con las nuevas tecnologías.

Esta aplicación es un proyecto conjunto de dos TFGs, por lo tanto, esta memoria se centrará en explicar la realidad aumentada dentro de la aplicación, aunque también se harán explicaciones sobre el desarrollo y funcionamiento en general para ponerla en contexto.

La realidad aumentada de la aplicación pretende que el usuario pueda acceder rápidamente toda la información correspondiente a cualquier medicamento. Para ello, solo se necesitará la cámara del dispositivo móvil enfocando a la caja de uno de ellos y tras la detección del medicamento será mostrada su información correspondiente. Todo esto se conseguirá gracias al uso combinado de tecnologías como OpenCV y OCR y el manejo de bases de datos.

En el siguiente documento se presenta toda la información necesaria para comprender el trabajo realizado. Se explicará detalladamente su investigación previa y cada una de las fases que se han seguido para el desarrollo del proyecto.

Palabras clave

Android, Realidad Aumentada, OpenCV, OCR, medicamento, Código Nacional, ARToolkit, base de datos, aplicación.

Abstract

Nowadays, there are many people with one or several diseases that require to follow a treatment with many medicines and many shots a day, and remembering when they have to take these shots could be complicated. This problem may even be worse for elderly people, because many of them have memory problems.

To help these people with their treatments, we are going to use smartphones, that nowadays have a high role in our daily lives and have come to these elderly people who did not use it before. Therefore, an application for Android mobile devices will be developed whose main focus is to help these people, because of this a clear and simple interface will be aimed to make the application user friendly for these elder people who are not familiar with new technologies.

This application is a combined project of two Bachelor Thesis, therefore, this document will focus on explaining the augmented reality within the application, although explanations of the development and general functioning will also be made to put this in context.

The augmented reality part of the application pretends that the user can quickly access all information about any medicine. To do this, it is only required the mobile device camera focuses on the box of one of them and after detecting the drug, the corresponding information will be displayed. All of this will be achieved through the combined use of technologies like OpenCV and OCR and database management.

In the next document, all the necessary information to understand the work done is presented. The previous research and each of the phases that have been followed for the development of the project will be explained in detail.

Keywords

Android, Augmented Reality, OpenCV, OCR, medicine, National Code, ARToolkit, database, application.

Agradecimientos

Tras terminar mi Trabajo de Fin de Grado, lo primero es dar las gracias a todas las personas que me han apoyado en los buenos y malos momentos, sin ellos ni este trabajo, ni el camino hacia él hubieran sido posibles.

A mi familia, a mis compañeros de carrera que ahora han acabado siendo amigos, al resto de amigos de la infancia y a todos los profesores que han aportado su granito de arena.

INDICE DE CONTENIDOS

| | | |
|---------|---|----|
| 1 | Introducción..... | 1 |
| 1.1 | Motivación..... | 3 |
| 1.2 | Objetivos..... | 3 |
| 1.3 | Fases del proyecto | 4 |
| 1.4 | Organización de la memoria..... | 6 |
| 2 | Estado del arte | 7 |
| 2.1 | Introducción..... | 7 |
| 2.2 | Realidad Aumentada..... | 7 |
| 2.2.1 | Tecnologías y aplicaciones de Nivel 0 | 8 |
| 2.2.2 | Tecnologías y aplicaciones de Nivel 1 | 8 |
| 2.2.3 | Tecnologías y aplicaciones de Nivel 2 | 9 |
| 2.2.4 | Tecnologías y aplicaciones de Nivel 3 | 10 |
| 2.3 | Herramientas y tecnologías usadas..... | 11 |
| 2.4 | Conclusiones..... | 12 |
| 3 | Diseño..... | 15 |
| 3.1 | Descripción del sistema | 15 |
| 3.1.1 | Descripción del sistema general | 15 |
| 3.1.2 | Descripción del sistema de realidad aumentada | 15 |
| 3.2 | Catálogo de requisitos | 15 |
| 3.2.1 | Requisitos funcionales..... | 15 |
| 3.2.1.1 | Requisitos de aplicación | 15 |
| 3.2.1.2 | Requisitos de usuario..... | 16 |
| 3.2.2 | Requisitos no funcionales..... | 16 |
| 3.2.2.1 | Requisitos de interfaz y usabilidad..... | 16 |
| 3.2.2.2 | Requisitos de documentación | 16 |
| 3.2.2.3 | Requisitos de recursos | 16 |
| 3.2.2.4 | Requisitos de rendimiento | 17 |
| 3.3 | Diagramas de Bases de datos..... | 17 |
| 3.3.1 | Base de datos de la aplicación | 17 |
| 3.3.2 | Base de datos del Vademécum | 19 |
| 3.4 | Diagramas de transición de estados..... | 20 |
| 3.5 | Maquetas..... | 22 |
| 4 | Desarrollo | 25 |
| 4.1 | Introducción..... | 25 |
| 4.2 | Funcionamiento general de la aplicación | 25 |
| 4.3 | Funcionamiento de la RA en la aplicación..... | 28 |
| 4.3.1 | Detección de bordes y componentes conexas..... | 28 |
| 4.3.2 | Identificación del medicamento | 29 |
| 4.3.3 | Seguimiento de objetos..... | 29 |
| 4.3.4 | Superposición de información | 30 |
| 5 | Integración, pruebas y resultados | 33 |
| 5.1 | Introducción..... | 33 |
| 5.2 | Pruebas | 33 |
| 5.2.1 | Pruebas generales de la aplicación | 33 |
| 5.2.2 | Pruebas con ARToolkit | 33 |
| 5.2.3 | Pruebas con OpenCV..... | 34 |
| 5.2.4 | Pruebas con OCR | 35 |

| | |
|---|----|
| 5.3 Resultados..... | 35 |
| 6 Conclusiones y trabajo futuro..... | 39 |
| 6.1 Conclusiones..... | 39 |
| 6.2 Trabajo futuro..... | 40 |
| Referencias | 41 |
| Anexos..... | I |
| A Manual del programador | I |
| B Tabla de tareas de prácticas en empresa..... | II |

INDICE DE FIGURAS

| | |
|---|----|
| FIGURA 1: CUOTA DE MERCADO ANDROID [1]..... | 1 |
| FIGURA 2: CUOTA DE VERSIONES ANDROID 2015 [2]..... | 2 |
| FIGURA 3: CUOTA DE VERSIONES ANDROID 2016 [2]..... | 2 |
| FIGURA 4: MODELO DE CICLO DE VIDA..... | 5 |
| FIGURA 5: EJEMPLOS DE CÓDIGOS USADOS EN EL NIVEL 0..... | 8 |
| FIGURA 6: EJEMPLO DE REALIDAD AUMENTADA..... | 9 |
| FIGURA 7: IMÁGENES DEL JUEGO INVIZIMALS..... | 9 |
| FIGURA 8: LAYAR IDENTIFICANDO UN MONUMENTO..... | 10 |
| FIGURA 9: APP ROUTE 66 MAPS..... | 10 |
| FIGURA 10: FUTURO DE LAS GOOGLE GLASS..... | 11 |
| FIGURA 11: DIAGRAMA DE LA BASE DE DATOS DE LA APLICACIÓN..... | 17 |
| FIGURA 12: DIAGRAMA DE TRANSICIÓN DE ESTADOS GENERAL DE LA APLICACIÓN..... | 21 |
| FIGURA 13: DIAGRAMA DE TRANSICIÓN DE ESTADOS DE LA REALIDAD AUMENTADA..... | 22 |
| FIGURA 14: MAQUETAS..... | 22 |
| FIGURA 15: PANTALLA PRINCIPAL DE LA APLICACIÓN..... | 25 |
| FIGURA 16: PANTALLA PARA RELLENAR LOS CAMPOS AL REGISTRAR UN MEDICAMENTO..... | 26 |
| FIGURA 17: PANTALLA DE LA FICHA DE UN MEDICAMENTO..... | 27 |
| FIGURA 18: FUNCIÓN PARA DETECTAR COMPONENTES CONEXAS..... | 28 |
| FIGURA 19: CÓDIGO DONDE SE OBTIENEN LAS COORDENADAS DE UNA CAJA..... | 30 |
| FIGURA 20: EJEMPLO DE SUPERPOSICIÓN DE INFORMACIÓN..... | 31 |

INDICE DE TABLAS

| | |
|--|----|
| TABLA 1: TAREAS Y HORAS | 6 |
| TABLA 2: TABLA MEDICAMENTOS, BASE DE DATOS VADEMÉCUM | 19 |
| TABLA 3: TABLA ENVASES, BASE DE DATOS VADEMÉCUM | 20 |
| TABLA 4: TABLA DE TAREAS DE PRÁCTICAS EN EMPRESA | II |

Acrónimos

CN: Código Nacional

OCR: Optical Character Recognition

OpenCV: Open Source Computer Vision

RA: Realidad Aumentada

Glosario

CN: Número único asignado a cada medicamento para facilitar su identificación.

OCR: Herramienta que permite la digitalización de textos. Dichos textos se identifican a partir de una imagen con caracteres y símbolos que pertenecen a un determinado alfabeto.

OpenCV: Es una biblioteca libre multiplataforma con más de 500 funciones para áreas de tratamiento de imágenes como el reconocimiento de objetos, que será usado en este proyecto.

1 Introducción

En los últimos años la cantidad de teléfonos móviles que existen en el mundo se ha incrementado sin parar, llegando a personas que hace poco no disponían de ellos. Empezaron siendo caros y exclusivos para empresarios y con el avance de la tecnología se han ido abaratando sus precios lo que ha conseguido que niños o ancianos accedan a ellos fácilmente. Actualmente están presentes en cualquier ámbito de la vida, personal o profesional, y con ello cualquier persona dispone de un pequeño ordenador que le facilita su vida diaria.

En este mercado de teléfonos móviles y con dicho aumento de dispositivos, la tecnología también se ha incrementado y diversificado llegando a tener varios sistemas operativos móviles en los terminales. En la actualidad, el sistema operativo móvil con más cuota de mercado mundial es Android, desde que se lanzó a finales de 2008 no ha hecho más que crecer como vemos en la siguiente imagen:

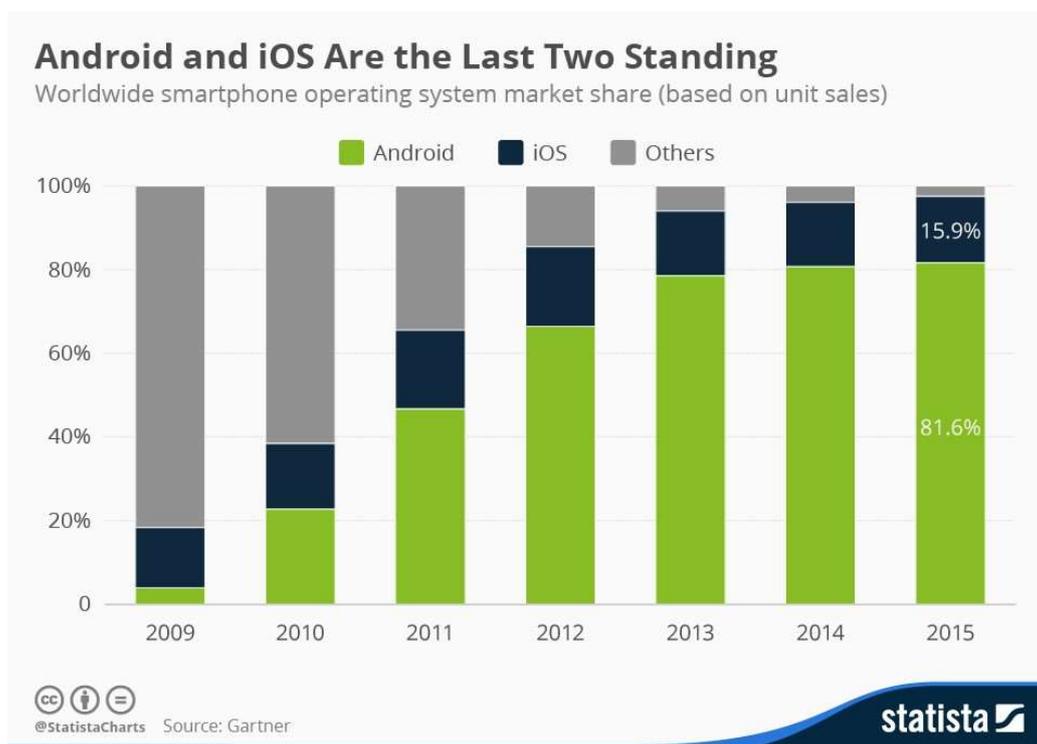
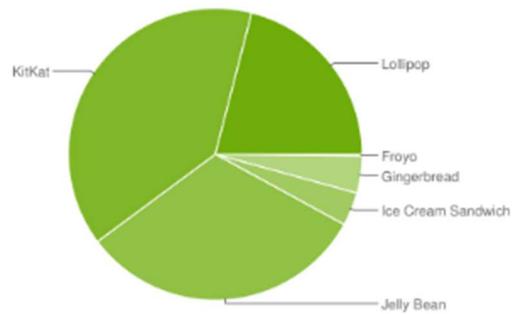


Figura 1: Cuota de Mercado Android [1]

Dentro del 80% de Android, existen otros porcentajes con sus distintas versiones. Cuando este trabajo de fin de grado se inició a principios de septiembre de 2015 las versiones predominantes en dispositivos Android eran Jelly Bean (versiones 4.1 a 4.3) y KitKat, (versión 4.4), tenían el 71% de

cuota de mercado. La versión Lollipop (5.0 y 5.1) estaba creciendo ya que el lanzamiento de la nueva versión 6.0, Marshmallow, era inminente.

| Version | Codename | API | Distribution |
|---------------|--------------------|-----|--------------|
| 2.2 | Froyo | 8 | 0.2% |
| 2.3.3 - 2.3.7 | Gingerbread | 10 | 4.1% |
| 4.0.3 - 4.0.4 | Ice Cream Sandwich | 15 | 3.7% |
| 4.1.x | Jelly Bean | 16 | 12.1% |
| 4.2.x | | 17 | 15.2% |
| 4.3 | | 18 | 4.5% |
| 4.4 | KitKat | 19 | 39.2% |
| 5.0 | Lollipop | 21 | 15.9% |
| 5.1 | | 22 | 5.1% |

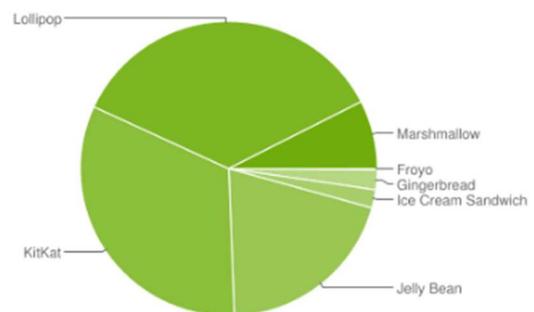


Data collected during a 7-day period ending on September 7, 2015.

Figura 2: Cuota de versiones Android 2015 [2]

Por ello, se decidió hacer nuestra aplicación compatible con las versiones KitKat y Lollipop (aunque también tiene bastante compatibilidad con Jelly Bean), ya que al terminarla serían las versiones más usadas dentro de Android como vemos en la gráfica, entre las dos suman casi el 70% de la cuota de mercado actual de Android.

| Version | Codename | API | Distribution |
|---------------|--------------------|-----|--------------|
| 2.2 | Froyo | 8 | 0.1% |
| 2.3.3 - 2.3.7 | Gingerbread | 10 | 2.2% |
| 4.0.3 - 4.0.4 | Ice Cream Sandwich | 15 | 2.0% |
| 4.1.x | Jelly Bean | 16 | 7.2% |
| 4.2.x | | 17 | 10.0% |
| 4.3 | | 18 | 2.9% |
| 4.4 | KitKat | 19 | 32.5% |
| 5.0 | Lollipop | 21 | 16.2% |
| 5.1 | | 22 | 19.4% |
| 6.0 | Marshmallow | 23 | 7.5% |



Data collected during a 7-day period ending on May 2, 2016.

Figura 3: Cuota de versiones Android 2016 [2]

1.1 Motivación

Hace unos pocos años cuando alguien enfermaba, iba al médico y le mandaban un tratamiento, lo primero que hacía era apuntar en un calendario hasta qué día tenía que estar tomando el medicamento y recordarlo cuando ese día llegaba. También cogía la caja y escribía para qué dolor o enfermedad era cada medicamento, ya que los nombres comerciales no suelen estar relacionados con las dolencias que trata cada uno. En la caja además era habitual apuntar la dosis que se debía tomar. Por último, el paciente tenía que recordar cuándo había que tomar la dosis, por ejemplo, cada 8 horas o con cada comida.

Actualmente, gracias al avance de la tecnología existen muchas aplicaciones para móviles de tipo calendario, que avisan cuándo debes tomar tu tratamiento o cuando acaba, pero nosotros queremos ir un paso más allá. Además de notificar las tomas personalizadas para cada medicamento se quiere tener más control sobre dichos tratamientos.

Y aunque la aplicación la podrá usar cualquier persona, su objetivo principal es ayudar a personas mayores. Muchos de ellos tienen uno o varios tratamientos con medicamentos debido a sus enfermedades. La motivación principal para desarrollar esta aplicación es querer ayudarles a seguir esos tratamientos y facilitarles todo lo posible sus tomas, evitando por ejemplo que se olviden de una toma, que se tomen el medicamento erróneo o a la hora incorrecta, o que finalicen su tratamiento antes o después de la fecha marcada.

1.2 Objetivos

El objetivo principal de este proyecto es la realización de una aplicación para móviles Android que ayude a seguir los tratamientos de medicamentos, orientada principalmente a personas mayores. El proyecto se ha realizado junto a otro alumno que presentará su propio TFG, llamado “Identificación de imágenes en dispositivos móviles Android para tareas interactivas”. Esta memoria, y por lo tanto los objetivos de este TFG se centrarán en la parte correspondiente a la realidad aumentada del proyecto conjunto, aunque también se describirán partes de la aplicación completa para entender mejor el contexto de este proyecto. Para ello, se deberán cumplir los siguientes objetivos:

- **Sencillez.** El uso de la aplicación debe ser lo más sencillo posible ya que muchos ancianos no están muy familiarizados con las nuevas tecnologías.
- La **interfaz** debe ser lo más **intuitiva** posible por lo expuesto en el punto anterior. Esto ayudará a que el usuario tome confianza rápidamente con la aplicación y facilite su uso continuo.

- La aplicación no debe fallar durante su uso, es decir, tiene que ser **fiable**. Es un objetivo bastante importante, ya que un fallo en la aplicación podría suponer la falta de seguimiento de un tratamiento.
- La detección de medicamentos mediante realidad aumentada debe ser **rápida**. Al igual que la fluidez en el seguimiento de las cajas tras su detección se acomodará a personas con mal pulso, como enfermos de Parkinson.
- La aplicación tendrá un **mantenimiento** durante el futuro, que irá en función de la experiencia con usuarios para adaptarla a sus necesidades.

1.3 Fases del proyecto

Llegado a este punto se van a explicar por las fases que ha pasado el proyecto

1. **Investigación y estudio de las tecnologías actuales.** Cualquier proyecto de software se inicia con un proceso de investigación, con ello se consigue recopilar información sobre la realidad aumentada para poder aplicarla a nuestra aplicación.
2. **Análisis.** Durante esta fase se definen los requisitos básicos para el correcto funcionamiento de la aplicación. Dichos requisitos marcarán los objetivos necesarios para completar la funcionalidad.
3. **Diseño.** Una vez definidos los requisitos se diseña la arquitectura de la aplicación con maquetas de las pantallas principales. También se realizan diseños de bases de datos y diagramas de transición de estados.
4. **Implementación.** Con los requisitos bien definidos y un buen diseño se puede empezar la implementación del código sabiendo que cumplirá los objetivos.
5. **Pruebas.** Aunque se realizan pruebas mientras se implementa el código, la mayoría de las pruebas vienen tras su finalización, para comprobar el correcto funcionamiento de la aplicación.
6. **Documentación.** Tras comprobar que la implementación es correcta y no hay ningún comportamiento inesperado, se pasa a redactar documentos donde se explica todo el trabajo realizado en el proyecto.
7. **Mantenimiento.** Es básico realizar una buena documentación para poder realizar este paso correctamente. Cerca de un 80% del ciclo de vida de un software es su propio mantenimiento, aunque para este proyecto no se sabe cómo se realizará.

Estas fases descritas forman el ciclo de vida usado en el proyecto. Se ha decidido usar un modelo en cascada porque es el que más se adecúa al desarrollo que se requería seguir para nuestra aplicación. La principal característica del modelo en cascada es que para empezar una fase se debe haber completado la fase anterior, con ello se garantiza un orden correcto a seguir. Se le ha añadido retroalimentación ya que en algún punto se ha tenido que acceder a fases anteriores debido a algún cambio por no haber completado los requisitos al no saber completamente hasta donde abarcaría el proyecto.

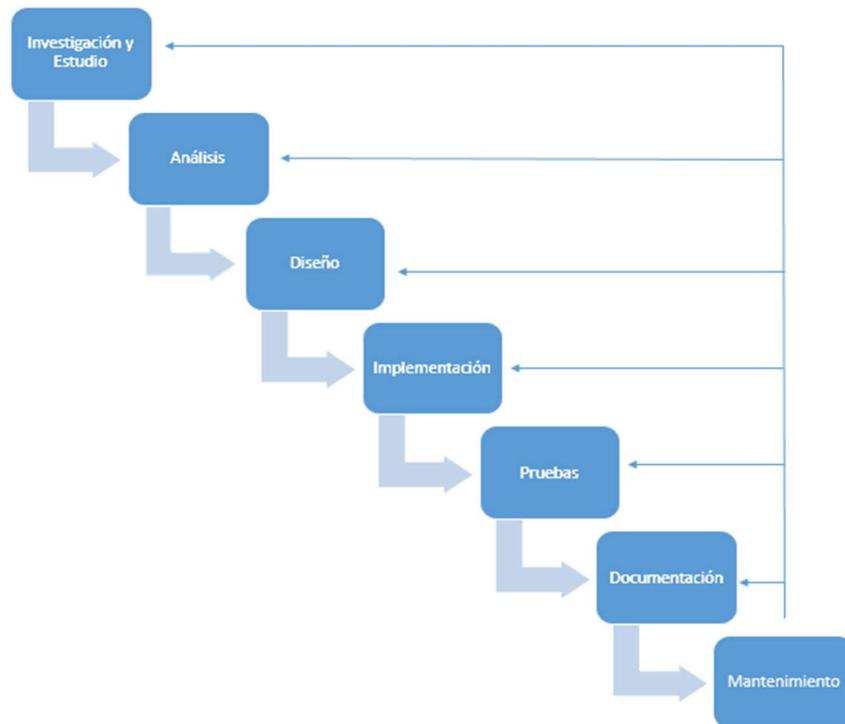


Figura 4: Modelo de Ciclo de Vida

Cabe destacar que en la fase de implementación se ha realizado más un desarrollo con modelo evolutivo. Esto es debido a que todo el proyecto se ha ido realizando basado en el modelo **Scrum** [3] con pequeños *sprints* de una o dos semanas. Estos *sprints* se iniciaban con una reunión entre todos los integrantes del proyecto para decidir el trabajo a realizar y la fecha de finalización, por ejemplo, buscar información, detallar los requisitos o crear la base de datos. En la siguiente reunión se hacía una puesta en común del trabajo realizado durante el *sprint* y se definía el siguiente. Por lo tanto, durante la fase de implementación se revisaba el análisis y diseño previamente hecho de un módulo o funcionalidad por si se necesitaba algún cambio, se desarrollaba y se hacían las pruebas pertinentes para comprobar su correcta integración, todo durante un mismo *sprint*.

Por último, se muestra en la siguiente tabla el tiempo dedicado a cada parte (y en el Anexo C se muestra la tabla de tareas realizadas en las prácticas en empresa asociadas a este TFG):

Tabla 1: Tareas y horas

| Tarea | Horas |
|---|-------|
| Investigación previa | |
| Familiarización las tecnologías OCR y OpenCV | 10 |
| Estudio e investigación de las tecnologías de realidad aumentada | 20 |
| | |
| Diseño | |
| Diseño de la base de datos (orientada a la realidad aumentada) | 5 |
| Diseño de la navegación por realidad aumentada | 20 |
| | |
| Desarrollo | |
| Implementación de ARToolkit en Android | 15 |
| Implementación de la navegación por realidad aumentada (con OpenCV y OCR) | 50 |
| | |
| Pruebas | |
| Pruebas con ARToolkit | 100 |
| Pruebas de la navegación por realidad aumentada (con OpenCV y OCR) | 80 |

1.4 Organización de la memoria

En este apartado se incluye un breve resumen del contenido de la memoria.

- **Capítulo 1.** En este capítulo se explican las motivaciones que han llevado a cabo la realización del proyecto, se exponen los objetivos a cumplir y se detallan las fases de desarrollo con el tiempo empleado en cada una.
- **Capítulo 2.** Aquí trataremos el tema central de nuestro proyecto. Se explicará que es la realidad virtual, en qué estado está actualmente y sus distintos tipos.
- **Capítulo 3.** En este apartado se empezará por detallar los requisitos funcionales y no funcionales de la aplicación. También se documentarán las decisiones de diseño tomadas junto a diagramas y maquetas que ayuden a planificar el proyecto.
- **Capítulo 4.** Tras explicar el diseño, se explicará todo lo relacionado con la implementación de la aplicación general y más detalladamente la parte que corresponde a este TFG, la Realidad Aumentada.
- **Capítulo 5.** Una vez realizada la implementación, se mostrarán las pruebas que han sido realizadas en la aplicación para garantizar su correcto funcionamiento.
- **Capítulo 6.** Por último, se explicarán las conclusiones sacadas sobre el trabajo realizado y se detallará brevemente el posible trabajo futuro a realizar sobre la aplicación.

2 Estado del arte

2.1 Introducción

Últimamente vemos cómo los avances tecnológicos están cada vez más presentes en diferentes sectores como educación, transportes, medicina, etc. Muchos de estos avances llegan para ayudarnos y facilitarnos nuestra vida cotidiana, como por ejemplo puede ser la realidad aumentada.

Según la definición de Wikipedia:

La Realidad Aumentada (RA) es el término que se usa para definir una visión a través de un dispositivo tecnológico, directa o indirecta, de un entorno físico del mundo real, cuyos elementos se combinan con elementos virtuales para la creación de una realidad mixta en tiempo real.

A menudo se puede confundir o no saber diferenciar la realidad aumentada de la realidad virtual. La principal diferencia entre ellas es que con la primera añadimos información virtual a la información física, mientras que con la segunda se crean escenas y objetos de apariencia real como, por ejemplo, en los videojuegos.

2.2 Realidad Aumentada

Si nos centramos en la realidad aumentada, Lens-Fitzgerald, cofundador de Layar de la cual se hablará más adelante, define 4 niveles [4] de complejidad según las tecnologías que usan:

- **Nivel 0 (Physical World Hyper Link):** Hipervínculo con el mundo físico. Las aplicaciones se enlazan a la realidad mediante códigos 1D y 2D (códigos de barras o QR, por ejemplo). Estos códigos solo son enlaces a otros contenidos. No hay registro de objetos en 3D. Hay gente que no considera este nivel como realidad aumentada.
- **Nivel 1 (Marked Based AR):** Realidad aumentada basada en marcadores. Se usan marcadores cuadrangulares con dibujos esquemáticos en blanco y negro. La forma más avanzada con este nivel permite reconocer objetos 3D.
- **Nivel 2 (Markerless AR):** Realidad aumentada sin marcadores. Las aplicaciones usan ahora el GPS, y la brújula del dispositivo móvil en lugar de marcadores para determinar la localización y orientación del usuario y por superponer puntos de interés a imágenes del mundo real tomadas con la cámara.

- **Nivel 3 (Augmented Vision):** Visión aumentada. Dispositivos de alta tecnología que ofrecen una experiencia totalmente inmersiva y personal, como lentes de contacto de alta tecnología o Google Glass que están siendo desarrolladas en la actualidad.

2.2.1 Tecnologías y aplicaciones de Nivel 0

Es la forma más antigua, por ello la más desarrollada y a la vez más simple. Comenzó con la identificación de códigos 1D (códigos de barras) para mostrar información de cualquier producto, luego avanzó al reconocimiento de códigos 2D (códigos QR).

Hay aplicaciones móviles para este nivel que escanean códigos 1D y 2D como son **Kaywa**, que desde su web nos permite generar un código QR a partir de una URL y su app es un lector y decodificador de estos códigos. **ShopSavvy** escanea el código de barras de un producto y nos busca tiendas físicas y online donde encontrarlo además de comparar precios, mostrar descuentos y mostrar opiniones de usuarios sobre dicho producto.

La aplicación más usada es la del reconocimiento de imágenes como el logo de una compañía, el cartel de una película o la carátula de un CD como hace **Kooba** y ser redirigido a una web para recibir más información. **TinEye** reconoce imágenes para hacer una búsqueda inversa o darnos imágenes similares. **Similify** permite encontrar ropa similar a partir de una foto también mediante el reconocimiento de dicha prenda de ropa.



Figura 5: Ejemplos de códigos usados en el nivel 0.

2.2.2 Tecnologías y aplicaciones de Nivel 1

Las primeras aplicaciones de este nivel estaban orientadas al PC y webcams, imprimir un marcador 2D en blanco y negro, sostenerlo delante de la cámara y ver una animación en 3D todo ello en tiempo real, esto lo hacían tecnologías como **Metaio** o **ARToolKit**.

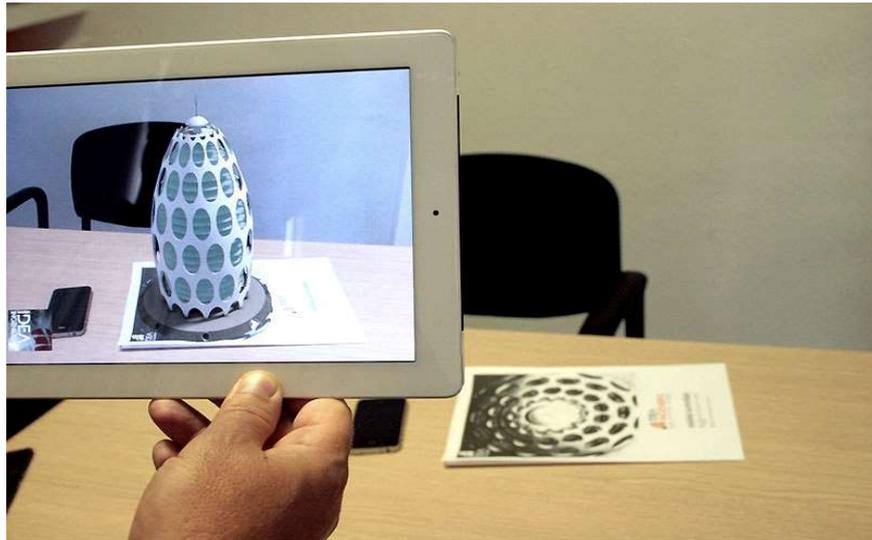


Figura 6: Ejemplo de realidad aumentada.

La mayoría de estas primeras aplicaciones ahora están orientadas a dispositivos móviles, incluidas **Metaio** y **ARToolkit**. Con **TotalImmersion** un usuario de la aplicación se puede probar distintos modelos de gafas virtuales o ver cómo quedaría un sofá o una estantería en una habitación.

Este nivel también se ha orientado hacia los videojuegos, como el juego **Invizimals** desarrollado por **Novarama** para PSP o el equipo de **int13** que desarrolla juegos de realidad aumentada para dispositivos móviles.



Figura 7: Imágenes del juego Invizimals.

2.2.3 Tecnologías y aplicaciones de Nivel 2

Las aplicaciones de este nivel están orientadas a móviles y suelen ser más usadas en exteriores al hacer uso principalmente del GPS y la brújula del dispositivo.

Wikitude o **Layar** muestran objetos en la cámara del móvil calculando nuestra posición por GPS o WiFi y la dirección con la brújula o el acelerómetro. Con esta tecnología se pueden mostrar el nombre y la distancia a puntos de interés cercanos como museos, monumentos, hospitales, restaurantes, tiendas, estaciones de metro, etc...



Figura 8: Layar identificando un monumento.

Otra utilidad muy usada en este nivel es la de navegación GPS, como **Route 66 Maps** o **Wikitude Drive** que muestran la ruta a seguir directamente sobreimpresa en las imágenes de la carretera que nos proporciona la cámara.



Figura 9: App Route 66 Maps.

2.2.4 Tecnologías y aplicaciones de Nivel 3

Actualmente no hay desarrollada demasiadas tecnologías para visión aumentada, como ya se ha dicho anteriormente se están desarrollando lentes de contacto de alta tecnología en las que se pueda montar una pantalla y mostrar información. El producto más avanzado hasta la fecha son las

Google Glass, que pueden ser controladas por comandos de voz y mostrar la información que se solicite en su pequeña pantalla.



Figura 10: Futuro de las Google Glass.

2.3 Herramientas y tecnologías usadas

Pasamos ahora a describir las herramientas y tecnologías que han sido utilizadas durante el desarrollo de todo el proyecto. Para ello, las clasificamos según su naturaleza, atendiendo a su relación con este trabajo:

Relacionadas con realidad aumentada

- **ARToolkit:** Biblioteca gratuita que permite la creación de aplicaciones de realidad aumentada. Utiliza las capacidades de seguimiento de video para calcular en tiempo real la posición de la cámara y la orientación de los marcadores físicos. [5]
- **OpenCV:** Otra biblioteca gratuita, en este caso contiene funciones para el tratamiento de imágenes, como el reconocimiento de objetos o facial, la calibración de cámaras o visión robótica, entre otras. [6]
- **OCR:** Reconocimiento óptico de caracteres por la traducción al castellano de sus siglas. Es una herramienta dirigida a la digitalización de textos. [7]

Relacionadas con dispositivos móviles

- **Android Studio:** Entorno gratuito de desarrollo integrado para realizar aplicaciones Android como la nuestra. Está basado en IntelliJ. [8]
- **API Android:** Como se ha justificado en una sección anterior, las API de Android usadas para el desarrollo de nuestra aplicación han sido Android 4.4 conocido como KitKat (Android 19) y Android 5.0 conocido como Lollipop (API 21).

Relacionadas con información médica (y gestión de la información en general)

- **Vademécum:** Es el catálogo de especialidades farmacéuticas, que contiene la totalidad de los medicamentos disponibles en el mercado organizados por grupos de indicaciones terapéuticas. Además, muestra información para cada medicamento como su composición, contraindicaciones, principio activo, etc. [9]
- **Datum Free:** Herramienta gratuita para gestionar bases de datos SQLite. Se ha usado para crear, modificar y hacer consultas a la base de datos de la aplicación.

Otras

- **BitBucket:** Servicio de alojamiento para proyectos que usan control de versiones Mercurial o Git. como es nuestro caso. Es muy útil el uso de repositorios cuando se trabaja en un proyecto conjunto. [10]
- **Trello:** Herramienta web gratuita para la gestión de proyectos. Usa un tablero con 3 listas de tareas por hacer, en proceso y terminadas donde se pueden crear tarjetas que representas estas tareas y añadir a sus integrantes, comentarios, prioridades, fechas, etc. [11]
- **Dropbox:** Servicio de alojamiento de archivos en la nube que se ha utilizado para no sobrecargar el repositorio cuando había que compartir documentación u otros archivos entre los integrantes del proyecto.
- **Microsoft Office:** Suite ofimática de Microsoft con distintas aplicaciones de escritorio. Se ha usado su editor de textos Word para escribir esta memoria y PowerPoint para la presentación del TFG.
- **ArgoUML:** Aplicación de diagramado escrita en Java, usada en este proyecto para montar los diagramas de transición de estados.
- **StarUML:** Otra herramienta para el dibujo de diagramas basada en estándares UML. Esta aplicación ha sido usada para el modelamiento del diagrama de la base de datos de la aplicación.

2.4 Conclusiones

En secciones previas se ha puesto en situación todo lo relacionado con la realidad aumentada que existe hoy en día, tras esto se puede decir que es una tecnología relativamente nueva y aunque ya tiene un gran número de aplicaciones en la vida diaria, también posee un gran potencial en aplicaciones futuras ya que muchas empresas están trabajando en ello.

Para nuestro proyecto se necesitaba una tecnología de Nivel 1, de las que usaban marcadores, por eso se decidió en un principio iniciar probando ARToolkit. Esta era la que más se ajustaba a las necesidades de nuestro proyecto y además era gratuita, no obstante, del resto de tecnologías de todos los niveles también se aprendieron ideas y conceptos que se acabaron aplicando a la aplicación que se desarrolló al final de este proyecto.

Como se puede observar también en el punto anterior se han utilizado una considerable cantidad de herramientas y tecnologías además de las relacionadas con la realidad aumentada. Al ser un proyecto grande, realizado por varias personas, se han usado varias herramientas para facilitar este trabajo en equipo como las de control de versiones o para la gestión de proyectos. El haber aprendido a manejar estas herramientas y participar en un proyecto en conjunto, me vendrá bien para mi futuro laboral. Por supuesto, también se han usado otras herramientas con las que ya estaba más familiarizado (por ejemplo, Microsoft Office), pero se han usado con más detalle, descubriendo funcionalidades nuevas que proporcionan y que, gracias al proyecto, he aprendido desde cero, además de mejorar el manejo que ya tenía previamente.

3 Diseño

3.1 Descripción del sistema

Antes de explicar los requisitos del proyecto, se harán una pequeña descripción de la aplicación general y otra de la realidad aumentada, para poder entender a qué hacen referencia dichos requisitos, así como para entender mejor el diseño propuesto del sistema.

3.1.1 Descripción del sistema general

El proyecto busca desarrollar una aplicación que permita al usuario registrar sus medicamentos y sea capaz de recordarle cuando tiene que tomar su tratamiento, calculando el número de pastillas restantes para avisarle también cuándo debe comprar más. El usuario en cualquier momento será capaz de acceder a la información de cada medicamento y editarla a su gusto.

Tanto para registrar el medicamento como para confirmar su toma, se realizará con un proceso automático de detección de objetos en imágenes (desarrollado en otro TFG).

3.1.2 Descripción del sistema de realidad aumentada

Este TFG está centrado en la realidad aumentada, por lo tanto, se va a explicar cómo aplicaremos esta tecnología a nuestra aplicación. Una de las funcionalidades de la aplicación será la navegación por la realidad aumentada, la cual mostrará información superpuesta a las imágenes de la cámara del dispositivo tras identificar un medicamento. Esta información variará en función de si el medicamento ha sido registrado en la aplicación por el usuario con su tratamiento o no, en caso de ser un medicamento registrado se podrá mostrar información de su tratamiento (próxima toma, pastillas restantes, etc.), en caso contrario, simplemente se mostrará información general sacada del Vademécum.

3.2 Catálogo de requisitos

3.2.1 Requisitos funcionales

3.2.1.1 Requisitos de aplicación

RF (1). - La aplicación será capaz de gestionar el registro de distintos tipos de medicamentos.

RF (2). - La aplicación será capaz de gestionar el acceso a la información de distintos tipos de medicamentos.

RF (3). - La aplicación tendrá un sistema de avisos al usuario.

Se notificará a los usuarios cuando deban tomarse su medicación.

Se notificará a los usuarios cuando su tratamiento vaya a finalizar.

RF (4). - La aplicación dispondrá de un historial con todas las tomas del usuario.

RF (5). - La aplicación tendrá una lista completa de los medicamentos registrados por el usuario.

3.2.1.2 Requisitos de usuario

RF (6). - El usuario deberá ser capaz de registrar el tratamiento de su medicación.

RF (7). - El usuario podrá editar y borrar el tratamiento de su medicación.

RF (8). - El usuario podrá acceder a una lista con todos sus medicamentos registrados.

Podrá ver la dosis que debe tomarse, la cantidad de pastillas restantes, el tiempo hasta la próxima toma y la información básica sobre ese medicamento disponible en el Vademécum.

RF (9). - El usuario podrá acceder a la información disponible en el Vademécum de cualquier medicamento (registrado o no) gracias a la Realidad Aumentada.

3.2.2 Requisitos no funcionales

3.2.2.1 Requisitos de interfaz y usabilidad

RNF (1). - Las pantallas de interacción de la aplicación se realizarán de forma que se permita un manejo completo, sencillo de aprender y utilizar por cualquier usuario, sea cual sea su nivel de conocimientos tecnológicos.

3.2.2.2 Requisitos de documentación

RNF (2). - La aplicación tendrá como idiomas disponibles castellano e inglés.

RNF (3). - La aplicación dispondrá de un apartado de ayuda (tutorial) que facilitará el manejo de la aplicación si se le plantea algún problema.

RNF (4). - La aplicación contará con un manual del desarrollador y un manual de usuario.

3.2.2.3 Requisitos de recursos

RNF (5). - La aplicación contará con un sistema de gestión de bases de datos para almacenar los registros de medicamentos y el historial de tomas.

3.2.2.4 Requisitos de rendimiento

RNF (6). - La aplicación será capaz de manejar un gran número de medicamentos registrados y sus correspondientes tratamientos.

RNF (7). - El tiempo de respuesta de la aplicación no deberá exceder de los 10 segundos.

3.3 Diagramas de Bases de datos

3.3.1 Base de datos de la aplicación

Nuestra aplicación requiere de una base de datos principal para su funcionamiento. En esta base de datos se almacenarán los medicamentos que el usuario registre previamente con sus respectivos tratamientos. Contará con las siguientes tablas y relaciones:

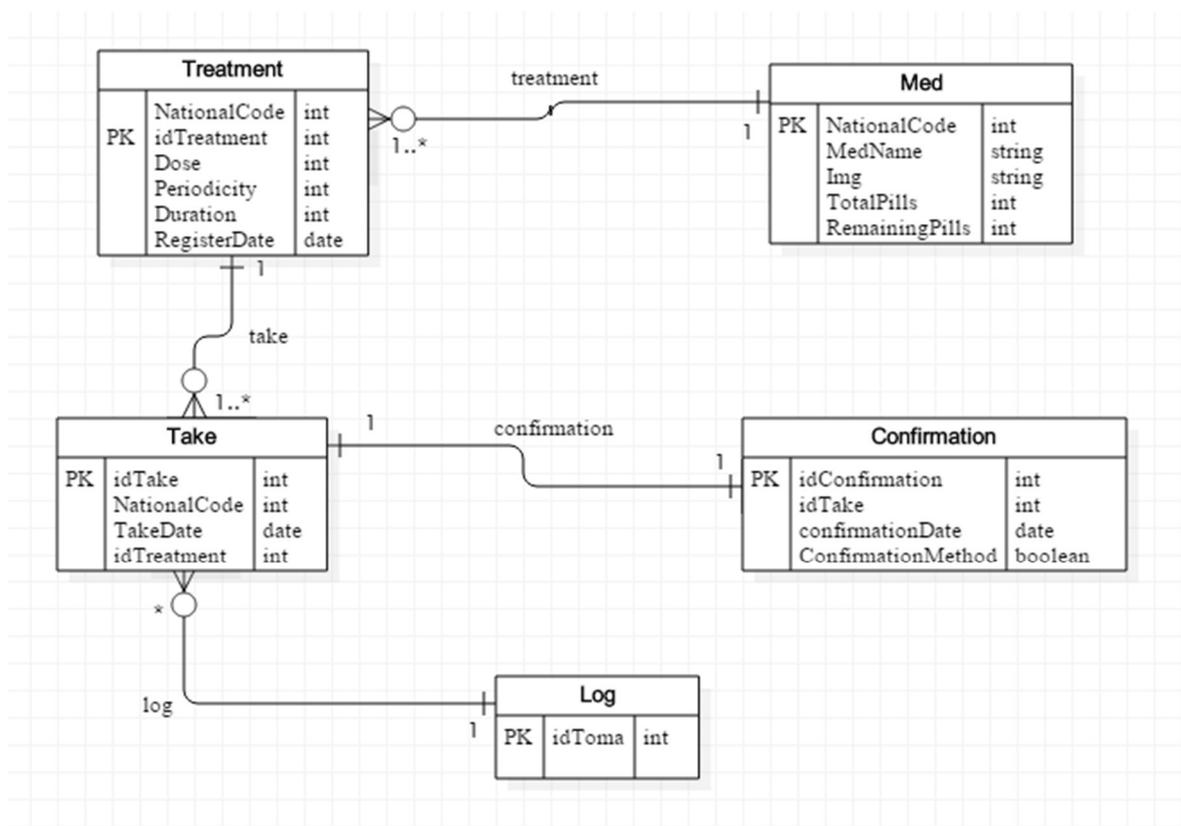


Figura 11: Diagrama de la base de datos de la aplicación

Como se puede ver la base de datos está formada por 5 tablas:

- **Tratamiento:** Contiene la información que añade el usuario al registrar un medicamento. Su clave primaria será un id autoincremental llamado *idTreatment*. El código nacional será clave foránea para relacionar el tratamiento con su medicamento en la tabla **Med**, esta será una relación 1-1, ya que cada medicamento podrá tener

activo un único tratamiento activo. La dosis será un tipo de datos *integer* para que se pueda almacenar un número de pastillas o mililitros de un medicamento líquido a tomar. En *periodicity* se guardará el tiempo entre tomas y en *duration* los días a seguir el tratamiento. En el campo *RegisterDate* se almacena la fecha de inicio del tratamiento y se podrá calcular el día de la última toma.

- **Medicamento:** El código nacional será la clave primaria de esta tabla ya que es un identificador único para cada medicamento y hará referencia a la tabla **Treatment** como se ha explicado más arriba. Además, contará con campos para guardar el nombre del medicamento, una foto de su caja que se usará para confirmar su correcta toma, el número de pastillas totales y el número de pastillas restantes para poder avisar al usuario cuando debe comprar una nueva caja.
- **Toma:** La tabla toma contará con un campo *idTake* que será su clave primaria autoincremental. Tendrá el código nacional del medicamento del cual es la toma y el identificador del tratamiento, que será una clave foránea para relacionarlo con la tabla **Tratamiento**. Esta relación será 1-* porque en un mismo tratamiento hay muchas tomas distintas. También existe un campo donde almacenar la fecha y hora en la que se debe realizar la toma para mostrar una notificación al usuario.
- **Confirmación:** En esta tabla se usará de nuevo una clave primaria autoincremental que será el identificador de la confirmación. También se dispondrá del identificador de la toma que se confirma y por lo tanto existirá una relación 1-1 entre esta tabla y la de tomas, a cada toma le corresponde una única confirmación. Se almacenará también la fecha en la que el usuario confirma que se ha tomado el medicamento y la forma en la que confirma, mediante la cámara o de forma manual.
- **Log:** De esta tabla se podrá sacar un historial de tomas del usuario. Se explicará detalladamente en el apartado de trabajo futuro, al final de la memoria, ya que no se ha desarrollado completamente.

Tras explicar la base de datos en general que usa al completo la aplicación, cabe destacar más detenidamente qué tablas se usarán para la realidad aumentada y cómo. Si un medicamento ha sido detectado por realidad aumentada mediante su código nacional, cuando se vaya a mostrar su información, se realizará una consulta a esta base de datos.

Más concretamente, se requerirán de la tabla **Med** los campos de pastillas totales y pastillas restantes para pintar al usuario una pastilla en función del porcentaje de las que queden. También se consultará a la tabla **Take** cuándo está programada la próxima toma de dicho medicamento

detectado, para así hacer la diferencia entre el resultado de la consulta y la hora actual y mostrar el tiempo restante hasta la toma.

3.3.2 Base de datos del Vademécum

Además de los medicamentos que el usuario pueda tener registrados, existen muchos más que aun así se podrán identificar mediante la realidad aumentada. Estos medicamentos estarán en la base de datos del Vademécum, la cual contiene todos los medicamentos disponibles. Se va a explicar brevemente esta base de datos y qué campos se usan ya que es importante en nuestro proyecto.

Esta base de datos está compuesta de dos tablas, por lo tanto, para mostrar sus campos y una explicación de cada uno de ellos se adjuntan a continuación dos tablas, correspondientes a cada una de las bases de datos:

Tabla 2: Tabla Medicamentos, base de datos Vademécum

| TABLA MEDICAMENTOS | |
|--|--|
| nom_med <<PRIMARY KEY>> | Nombre completo del medicamento |
| alertas_comp | Alertas por composición relacionadas con el embarazo, lactancia, conducción, etc. |
| ATC | El código ATC o Sistema de Clasificación Anatómica, Terapéutica, Química (ATC: acrónimo de Anatomical, Therapeutic, Chemical classification system) es un índice de sustancias farmacológicas y medicamentos, organizados según grupos terapéuticos. |
| PA | Principio activo |
| EXC | Excipientes |
| conservar_frio | Sí o No |
| tratamiento | Enfermedades o dolencias para las que se usa |
| que_es | Qué es el medicamento |
| antes_de_usar | Antes de usar el medicamento |
| como_tomar | Cómo tomar el medicamento |
| efectos_adversos | Efectos adversos/secundarios que puede tener el medicamento |
| conservacion | Cómo conservar el medicamento |
| composicion | Composición |
| contenido | Cuántas dosis tiene y en qué forma se presenta |

Tabla 3: Tabla Envases, base de datos Vademécum

| TABLA ENVASES | |
|--|--|
| CN <<PRIMARY KEY>> | Código nacional |
| nom_med <<FOREIGN KEY>> | Nombre del medicamento completo (clave primaria de la tabla “medicamentos”) |
| tipo_env | Tipo de envase, es decir, cuántas dosis contiene, de cuantos mg, etc. |
| SNS | Sistema Nacional Sanitario. Vale 1 si lo cubre la seguridad social y 0 si no o no se tiene datos sobre ello. |
| comercializado | 1 si se comercializa y 0 en caso contrario. |
| situacion | Alta o Anulado |
| PVL | Precio del laboratorio |
| PVPiva | Precio de venta |
| Pfacturacion | Precio facturación |
| EFG | 1 si es genérico y 0 si no lo es o no se tienen datos de ello. |
| H | 1 si es hospitalario y 0 si no lo es o no se tienen datos de ello. |
| EFP | 1 si es publicitario y 0 si no lo es o no se tienen datos de ello |

Como se puede observar, en esta base de datos hay muchos detalles que se podrían mostrar en la capa de información que se pinta tras detectar un medicamento. Por nuestra parte, se ha decidido no sobrecargar mucho la pantalla con demasiados iconos o texto de información.

Por lo tanto, lo único que se va a mostrar actualmente son las alertas por composición. Estas alertas son para evitar que ciertas personas tomen el medicamento, como pueden ser: mujeres embarazadas o lactantes, deportistas por riesgo de que el medicamento cuente como *dopping*, personas con problemas de fotosensibilidad o trabajadores que vayan a manejar maquinaria o vehículos pesados.

Para llegar a esta información, se realizará una consulta a la base de datos sabiendo el código nacional del medicamento del cual queremos obtener sus alertas por composición. Se obtendrá su nombre, que es un campo con clave foránea, de la tabla **Envases** y con ello se llegará a la tabla **Medicamentos**.

3.4 Diagramas de transición de estados

Se van a mostrar y explicar a continuación varios diagramas de transición de estados sobre la aplicación en general y sobre la parte de la RA más detalladamente:

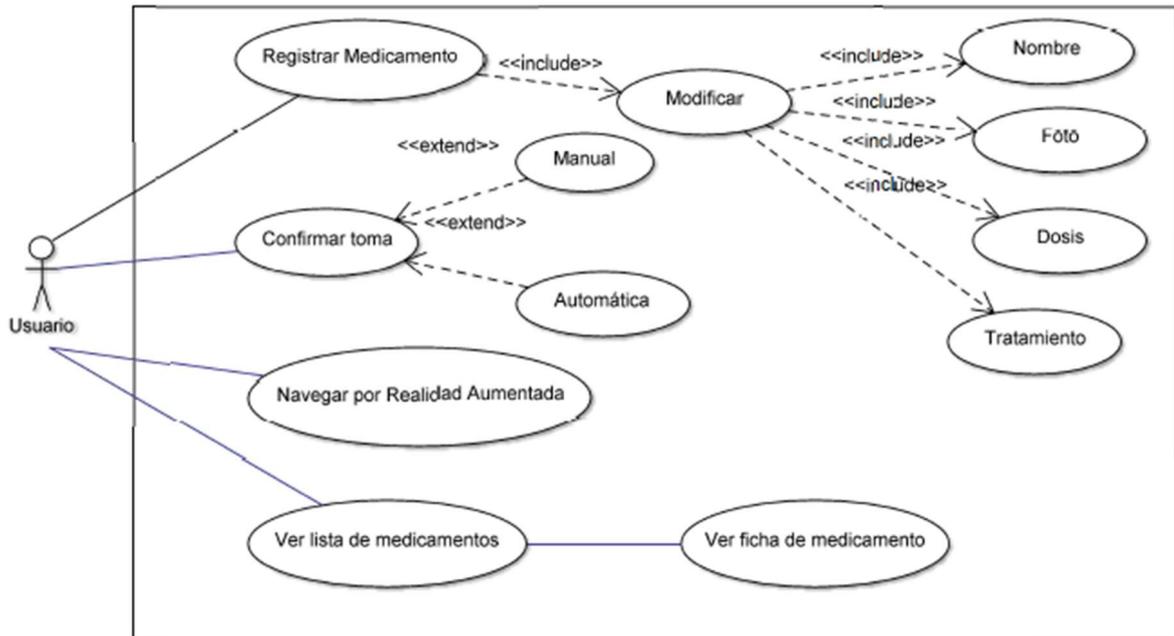


Figura 12: Diagrama de transición de estados general de la aplicación

Las acciones básicas que puede realizar el usuario son cuatro: registrar un medicamento, confirmar una toma, navegar por la realidad aumentada y ver la lista de medicamentos que tiene registrados. Estas 4 acciones serán 4 botones en la aplicación que mostraremos y detallaremos más adelante en el apartado 4.2 cuando expliquemos el funcionamiento de la aplicación.

Al registrar un medicamento, se incluye la opción de poder modificar más adelante, y con ello se puede editar sus campos: nombre, foto, dosis, etc. Confirmar una toma requiere que sea de una forma manual o automáticamente con la cámara del móvil. Y al ver la lista de medicamentos se mostrarán todos los medicamentos registrados, con la opción de poder acceder a la ficha de cada uno de ellos para poder ver su información.

Por último, se puede iniciar la navegación por la RA, que detallamos mejor con el siguiente diagrama:

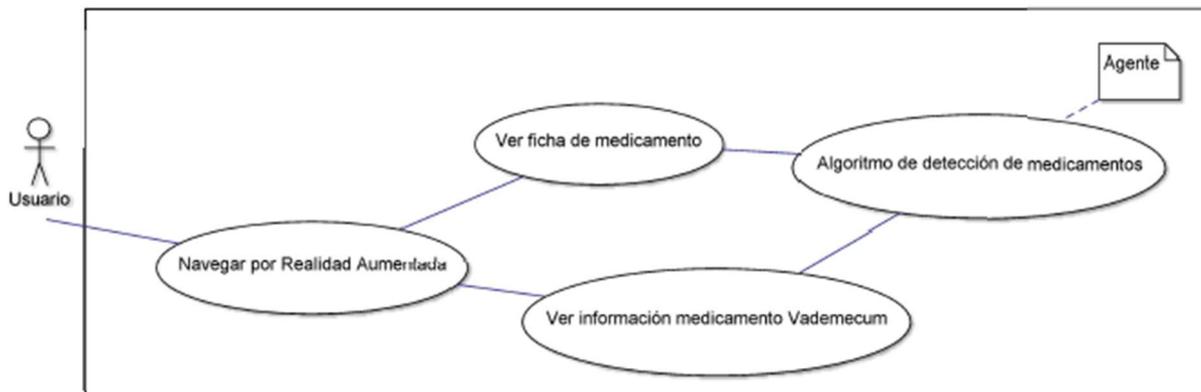


Figura 13: Diagrama de transición de estados de la Realidad Aumentada

Mientras se navega por la RA y se identifica un medicamento mediante el algoritmo de detección (es un agente) se puede acceder a la ficha de dicho medicamento, pero distinguiendo dos “tipos” de ellos, los que han sido registrados previamente por el usuario y están en la base de datos interna de la aplicación y los medicamentos no registrados pero que podemos acceder a su información gracias a la base de datos del Vademécum.

3.5 Maquetas

Antes de iniciar la implementación de la aplicación se diseñaron unas maquetas de la interfaz para guiarnos. Se quería seguir la normativa de diseño de Google para las nuevas versiones de Android, **Material Design** [12], por lo tanto, se investigaron algunas aplicaciones famosas que seguían este patrón para tomar ideas y las ajustamos a nuestra necesidad. Con ello se pintaron las siguientes maquetas para las pantallas más importantes de la aplicación:

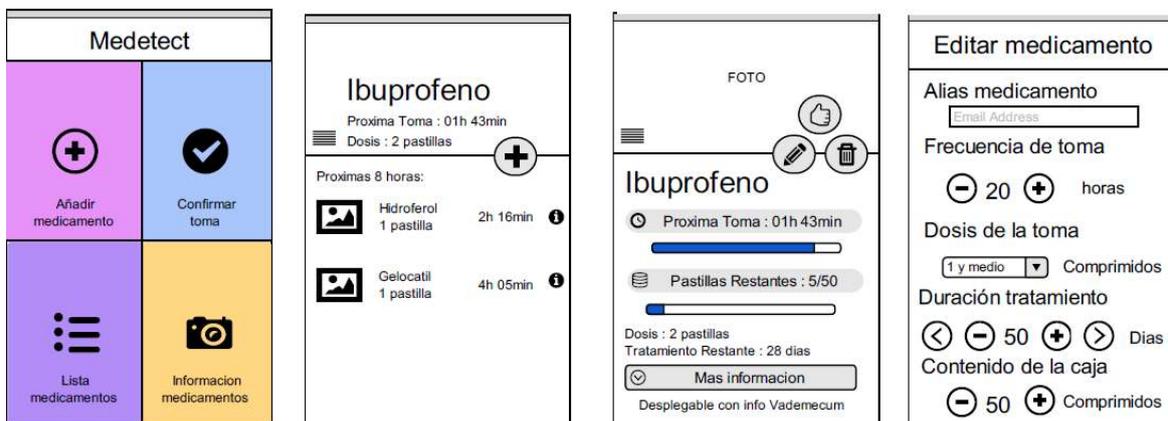


Figura 14: Maquetas

Como se observa, desde un primer momento se tenía claro el diseño de la pantalla principal, con unos botones grandes y de colores llamativos. El resto de pantallas se diseñaron con la idea de seguir Material Design y se consiguió mantener hasta el final, pero sufrieron algunos cambios durante el desarrollo, la mayoría para simplificar su interfaz y hacerla más clara. En una sección posterior se mostrará el resultado final de alguna de estas pantallas.

4 Desarrollo

4.1 Introducción

Nuestro proyecto es una aplicación para Android que busca el poder ayudar a cualquier persona a seguir el tratamiento de un medicamento. En esta sección se va a explicar cómo se va a conseguir esto brevemente, pero como esta memoria se centra en la parte de realidad aumentada, se explicará con más detalle dicha parte.

Como ya se ha explicado anteriormente, la idea base de realidad aumentada para nuestra aplicación es la de superponer información en vivo sobre el medicamento que el usuario enfoca con la cámara de su móvil.

4.2 Funcionamiento general de la aplicación

Antes de entrar a explicar con detalle la realidad aumentada, que es la parte principal de este TFG, se necesita dar una explicación clara del funcionamiento completo de la aplicación para poner todo ello en contexto.

Para explicar dicho funcionamiento, se van a aprovechar capturas de pantalla tomadas de la aplicación, por ello se empieza mostrando la primera pantalla que se muestra al usuario tras arrancar la aplicación:



Figura 15: Pantalla principal de la aplicación

Como se puede apreciar, es una pantalla con cuatro botones, en los que cada uno de ellos tiene una de las funcionalidades básicas de la aplicación. Esta pantalla principal busca la sencillez de cara a los usuarios a los que está orientada la aplicación, personas mayores que no están demasiado familiarizadas con las nuevas tecnologías. Por ello, se crearon los cuatro botones de gran tamaño, ajustándolos al máximo que nos proporciona la pantalla en el dispositivo que se lance la aplicación, con colores llamativos y bien diferenciados y letras e iconos grandes y fáciles de interpretar su significado.

Desde la pantalla principal, si se pulsa el botón naranja para agregar un medicamento se mandará al usuario a un proceso por el cual deberá hacer una foto a la caja del propio medicamento que desee registrar y tras ello se le mostrará la siguiente pantalla con los campos a rellenar:

← Registrar medicamento

Nombre del medicamento

Frecuencia de la toma
Seleccione la frecuencia con la que debe tomar el medicamento

Dosis de la toma
Seleccione la dosis del medicamento para cada toma

Fin del tratamiento
Seleccione el día que finaliza su tratamiento

Contenido de la caja
0 pastillas o sobres

Modificar foto de la caja
Actualizar la foto del medicamento

Figura 16: Pantalla para rellenar los campos al registrar un medicamento

No se va a entrar en demasiados detalles en este proceso, ya que corresponde más al otro TFG asociado a este proyecto, simplemente merece la pena destacar que, mediante la misma tecnología de OCR que se usará en nuestro caso para la realidad aumentada, de la foto tomada antes de llegar a esta pantalla también se puede obtener el código nacional del medicamento y autocompletar

algunos de los campos mostrados, dándole al usuario siempre la posibilidad de editarlos o cambiarlos antes de terminar de registrar el medicamento con su tratamiento, y más adelante también.

Si se vuelve a la pantalla principal y se pulsa el botón verde para confirmar la toma de un medicamento se abrirá de nuevo la cámara para hacer una foto a la caja y compararla con la foto que se guardó al registrar el medicamento. Esto se realizará mediante la comparación de imágenes. De nuevo, esta parte corresponde al otro TFG por lo que no se explicarán más detalles en esta memoria.

Al usuario también se le ofrece una lista con todos los medicamentos que ha registrado, a ella puede acceder con el botón azul disponible en la pantalla principal. Desde esa lista, el usuario puede pinchar sobre cualquier medicamento y navegar por su ficha de información:



Figura 17: Pantalla de la ficha de un medicamento

En esta ficha, como se puede ver, se mostrará la foto que tomó el usuario al registrar el tratamiento, el tiempo restante hasta su próxima toma, las pastillas restantes en el recipiente y la

dosis que se debe tomar. Además, como se ha dicho, con el botón naranja podrá editar toda esta información, también borrar el registro o dejar el medicamento en estado inactivo, en caso de que haya terminado su tratamiento, pero quiera mantenerlo registrado para el futuro.

Para terminar, en la pantalla principal el último botón que se muestra es el que abre la navegación por realidad aumentada, que se explica en el siguiente punto.

4.3 Funcionamiento de la RA en la aplicación

En esta sección se pasa a describir de una manera detallada el funcionamiento de la Realidad Aumentada de nuestra aplicación, que es el propósito general de este trabajo.

Como explicaremos más adelante, en el apartado de pruebas, ARToolkit y otras herramientas se descartaron porque no ayudaban a cubrir la funcionalidad que se requería. Por lo tanto, se propuso desarrollar nuestra propia RA mediante: detección de bordes y componentes conexas, la identificación del medicamento usando OCR, seguimiento de objetos y superposición de información gracias utilizando OpenCV.

```
//do_connectedcomponents
Mat bw = new Mat();
double threshval = 80;
Imgproc.threshold(grayInnerWindowReduced, bw, threshval, 0,
    threshval < 128 ? Imgproc.THRESH_TOZERO : Imgproc.THRESH_TOZERO_INV);
final Mat labelImage = new Mat(bw.size(), CvType.CV_32S);
Mat stats = new Mat();
Mat centroids = new Mat();
int nLabels = Imgproc.connectedComponentsWithStats(bw, labelImage, stats, centroids);
```

Figura 18: Función para detectar componentes conexas

Antes de todo lo que se va a explicar a continuación se instalaron y probaron las tecnologías OpenCV y OCR, que también será explicado debidamente más adelante, en la sección de pruebas junto a las realizadas sobre ARToolkit.

4.3.1 Detección de bordes y componentes conexas

Para identificar un medicamento lo primero que se necesita es detectar la caja independientemente del medicamento que sea, ya que es la parte común entre todos ellos, sean pastillas, cápsulas, viales o botes de líquido, todos ellos están dentro de una caja. Esta primera parte se realizará gracias a OpenCV y algunos de los métodos de tratamientos de imágenes que nos facilita.

Al recoger un *frame* de la cámara del móvil, lo primero que se realiza con el objetivo de reducir el tiempo de procesamiento es pasar la imagen a blanco y negro, un *smooth* (suavizado) que

ayudará a reducir el ruido en la imagen y se termina por aplicar una reducción del tamaño que facilitará todo el manejo futuro. Aun así, el *frame* original se guarda para su uso más adelante.

Una vez se tiene una imagen más fácil de manejar, se aplica un método de OpenCV que devuelve las componentes conexas que encuentra en una imagen. Dicho método, que se mostrará a continuación, lo que hace es identificar todos los bordes que existen en una imagen y además agrupa los que interpreta que pertenecen al límite de un mismo objeto, devolviendo así en una matriz (OpenCV trata las imágenes como matrices) cada componente conexas identificadas por una etiqueta.

Como se ve en la imagen superior, el método *connectedComponentsWithStats* nos devuelve el número de etiquetas distintas que ha asignado para cada componente. En su segundo argumento, la matriz *labelImage*, guarda la imagen con cada pixel cambiado por la etiqueta de la matriz a la que pertenece. Su primer argumento, *labelImage*, es por tanto la imagen de entrada de la cual queremos obtener las componentes conexas que existen en ella. Y el último argumento que nos interesa para nuestra explicación es el tercero, *stats*, que es donde se guarda información que se explicará más adelante porque es útil y como se usa.

Con esta matriz de etiquetas, *labelImage*, se puede identificar la componente más grande. Contando la etiqueta que más se repite sabemos cuál es, con ello se pasa a ignorarla (ya que en teoría debería ser el fondo y no nos interesa). Lo que se hace entonces es guardar la segunda etiqueta más grande que *debería ser* la caja del medicamento.

4.3.2 Identificación del medicamento

Llegados a este punto, se recupera el *frame* original para aplicar OCR en la parte correspondiente a la componente conexas de la imagen original, con ello se ahorra procesamiento y tiempo de nuevo.

El método de OCR devuelve un *array* con las palabras que ha detectado. Con cada una de esas palabras se comprueba si es un número y, en ese caso, se mira si ese número está comprendido en el intervalo de 150000 y 999999, si es afirmativo se tiene un posible Código Nacional y por lo tanto un medicamento identificado.

Tras identificar el CN se consultará primero en la base de datos de medicamentos registrados por el usuario para saber si más adelante debemos sacar la información que se va a mostrar de la base de datos de la aplicación o del Vademécum.

4.3.3 Seguimiento de objetos

Para el seguimiento de la caja se debe volver a la matriz de etiquetas que se obtenía del método que identificaba componentes conexas. Dicho método, se ha dicho que además proporciona unos

stats de cada componente que detecta, como pueden ser su coordenada más alta y su coordenada más a la izquierda. Gracias a estas coordenadas que se obtienen de cada *frame* que se procesa se tienen puntos donde se podrá situar la información que queremos mostrar de dicho medicamento.

En la siguiente imagen se muestra cómo se obtienen estas coordenadas gracias a los *stats* que nos proporciona la matriz. El código pertenece a un bucle en el que se realizan más acciones, pero no se muestra entero, solo la parte que interesa para esta sección. Todo ello se hace, como se ha dicho anteriormente, tras ignorar la primera etiqueta más repetida:

```
// take most frequent connected components (ignoring 0, the background)
for (int i = startingLabel; i < Math.min(topComponents + startingLabel,
    listLabelSizes.size()); i++) {
    if (label == listLabelSizes.get(i).getKey()) {
        flag = true;
        coordLeft = stats.get((label), Imgproc.CC_STAT_LEFT)[0];
        coordTop = stats.get((label), Imgproc.CC_STAT_TOP)[0];
```

Figura 19: Código donde se obtienen las coordenadas de una caja

Antes de ver que se podían obtener estos *stats* se iba a hacer un seguimiento del objeto por colores, lo cual es más complicado de implementar y más costoso para la aplicación, lo que implicaría más tiempo tanto de desarrollo como de procesamiento.

4.3.4 Superposición de información

Llegados a este punto donde se tiene el medicamento detectado y donde está situado, lo último que se necesita es mostrar la información del medicamento sobre la imagen recibida de la cámara. Para ello, se obtiene dicha información de la base de datos correspondiente, la base interna si el medicamento fue registrado previamente por el usuario o la base del Vademécum si es un medicamento externo a nuestra aplicación, y se superpone en las coordenadas obtenidas anteriormente gracias a los *stats* sobre un nuevo *layout* de Android.

Para mostrar cómo se maneja este *layout*, se muestra en la imagen una parte del código en el que se muestra cómo se pinta un pequeño botón de información que es común tanto a los medicamentos registrados en la base de datos de la aplicación, como a los externos que obtendremos de la base de datos del Vademécum. Este icono, tendrá además funcionalidad de botón, lo que causará que al ser pulsado nos mande a la ficha detallada de un medicamento:

```

imageButtonInformation.setOnClickListener((arg0) -> {
    Intent gotoVademecum = new Intent(ImageManipulationsActivity.this,
        Vademecum.class);
    gotoVademecum.putExtra(Vademecum.MED_NC_PARAM, finalNc);
    startActivity(gotoVademecum);
});

runOnUiThread(() -> {
    imageButtonInformation.setVisibility(View.VISIBLE);
    RelativeLayout.LayoutParams layoutParams = (RelativeLayout.LayoutParams)
        imageButtonInformation.getLayoutParams();
    layoutParams.leftMargin = (finalCoordLeft.intValue() * 4); /**4 resize back
    layoutParams.topMargin = (finalCoordTop.intValue() * 4); /**4 resize back
    layoutParams.rightMargin = 0;
    layoutParams.bottomMargin = 0;
    imageButtonInformation.setLayoutParams(layoutParams);
});

```

Figura 20: Ejemplo de superposición de información

Como se puede observar, *finalCoordLeft* y *finalCoordTop* son las coordenadas que se ha explicado cómo se obtienen en el punto anterior y corresponden a las variables *coordLeft* y *coordTop* de la imagen de ese punto.

La información que se mostrará será distinta si el medicamento está registrado en la base de datos de la aplicación por el usuario o no. En el caso de un medicamento registrado se informará con iconos de las pastillas restantes en el envase o del tiempo hasta su próxima toma. En cambio, si el medicamento no ha sido registrado, sólo se mostrarán mediante iconos sus alertas por composición sacadas de la base de datos del Vademécum, que se ha explicado en la sección 3.2.2. Algunos ejemplos de estos casos serían alertas para que dicho medicamento no lo tomen mujeres embarazadas o lactantes, personas con fotosensibilidad o gente que vaya a manejar vehículos pesados o maquinaria.

5 Integración, pruebas y resultados

5.1 Introducción

En este capítulo se van a presentar detalladamente las pruebas realizadas sobre el conjunto de la aplicación desarrollada, explicando con más carga las realizadas sobre la parte de la realidad aumentada. Tras dichas pruebas se van a mostrar varias capturas de pantalla tomadas sobre la aplicación para mostrar su funcionamiento.

5.2 Pruebas

5.2.1 Pruebas generales de la aplicación

Aunque no es el objetivo principal de este TFG, hay que mencionar que se han realizado pruebas sobre la funcionalidad total de la aplicación, más allá de la realidad aumentada que se explica en esta memoria y la parte correspondiente al otro TFG.

Por ejemplo, con la base de datos se realizaron sus pruebas pertinentes para comprobar su correcto funcionamiento y tras su integración más pruebas para ver su buena comunicación entre base de datos y aplicación. También con todos los módulos que se fueron incluyendo se realizaron sus pruebas de integración correspondientes, como ya se ha comentado anteriormente. Para no entrar en excesivos detalles, se realizaron pruebas básicas como pueden ser probar el funcionamiento de todos los botones, que la navegación entre pantallas era la requerida o ajustar la interfaz para su correcto visionado en otros dispositivos con pantallas de distinto tamaño como tablets.

Para finalizar, también cabe destacar que se han tenido en cuenta pruebas realizadas con usuarios de cara a buscar la máxima sencillez para la interfaz.

5.2.2 Pruebas con ARToolkit

Tras analizar el catálogo de tecnologías sobre RA existentes, ARToolkit fue la que más se adecuaba a nuestra necesidad en un principio, por lo tanto, fue la elegida para empezar a investigar y desarrollar con ella nuestra aplicación.

Esta tecnología comenzó desarrollándose para PCs, por lo que las primeras pruebas se realizaron en un MacBook al ser más fácil su instalación que directamente en un dispositivo móvil. Estas primeras pruebas, con ejemplos de programas que proporciona el mismo ARToolkit con su SDK, simplemente consistían en familiarizarse con la tecnología y ver su potencial, comprobar cómo funcionaba el tracking de los marcadores y descubrir si podría acabar sirviéndonos para nuestro propósito final.

Tras comprobar que dicha tecnología en ordenadores funciona muy bien y podía servirnos, pasamos a su versión para Android. Se instalaron las librerías y cumplieron los demás requisitos que necesitaba ARToolkit para funcionar en móviles Android. Con ello se consiguió realizar las mismas pruebas que en la versión de PC.

En estas primeras pruebas en móvil se comprobó que ARToolkit es realmente rápido y fiable detectando sus marcadores incluso para la menor potencia que tiene un móvil comparado con un PC, por lo que la siguiente idea fue crear nuestros propios marcadores únicos para cada medicamento. Como se ha dicho estos marcadores serían únicos, ya que llevarían el código nacional de cada medicamento al igual que en la caja. Para crearlos se iba a usar un programa que proporciona el SDK e intentar automatizar el proceso de creación, pero era un proceso muy complicado y costoso y los resultados de identificar estos marcadores generados eran bastante malos. Por lo tanto, se desechó esta técnica para nuestro objetivo.

Lo último que se probó fue intentar que la aplicación detectara las cajas de medicamentos sin ningún tipo de marcador, cosa que en nuestro escenario tampoco es posible ya que la detección de objetos por imagen de ARToolkit requiere que en la imagen que se va a analizar no prevalezca un solo color, si no que existan muchos cambios de tonalidades, lo cual no sirve para identificar una caja de un medicamento en la que generalmente destaca un color (blanco en la mayoría de ellas) con algunas franjas y letras en otro color.

Llegados a este punto en el que fracasó nuestra primera idea para la realidad aumentada se hizo otro repaso al resto de tecnologías de realidad aumentada que hay en el mercado, pero todas se volvieron a descartar como un primer momento, ya que o bien eran tecnologías privadas de pag, no terminaban de acomodarse a nuestra necesidad o no cumplían alguno de los requisitos que se requerían. Por ello, se acabó proponiendo el desarrollo de nuestra propia tecnología, mediante el uso de OpenCV y OCR, como se ha explicado en el punto 4.3.

5.2.3 Pruebas con OpenCV

Igual que con ARToolkit, OpenCV proporciona unos ejemplos junto a las librerías que hacen falta para usar esta tecnología. Tras instalar dichas librerías requeridas, se empezó por probar un programa de ejemplo en el que se trataban de forma diferente las imágenes obtenidas por la cámara, mostrándolas en blanco y negro o en color sepia, por ejemplo.

En este programa también se nos proporcionaba un ejemplo de detección de bordes, con el que se empezó a probar si mediante OpenCV se detectaban correctamente los bordes de las cajas de los medicamentos. Comprobando que tras un tratamiento previo de la imagen como el suavizado y hacer un poco más pequeña la imagen, se ajustaba a lo que se quería.

Además de la detección de bordes, se probó también el funcionamiento de la detección de componentes conexas, que sobre el papel podría ajustarse más a lo que requería nuestra aplicación. Su funcionamiento ya ha sido explicado en la sección anterior, donde se explica cómo funciona toda la navegación por la realidad aumentada, por lo tanto, se quedó como el método que más se ajustaba y el usado para el desarrollo final de la realidad aumentada.

Tras decidir que se iba a usar esta función de detección de componentes conexas en la aplicación se realizó mucho trabajo previo para comprobar su correcto funcionamiento y ajustar todos sus parámetros a nuestras necesidades. Más adelante, en la sección 5.3 donde mostramos los resultados se hablará más sobre esta función.

5.2.4 Pruebas con OCR

Un método similar se usó para las pruebas sobre OCR. Se hizo un trabajo de investigación sobre otras aplicaciones que usaban esta tecnología para ver que podía ser usado y cómo acomodarlo en nuestra aplicación.

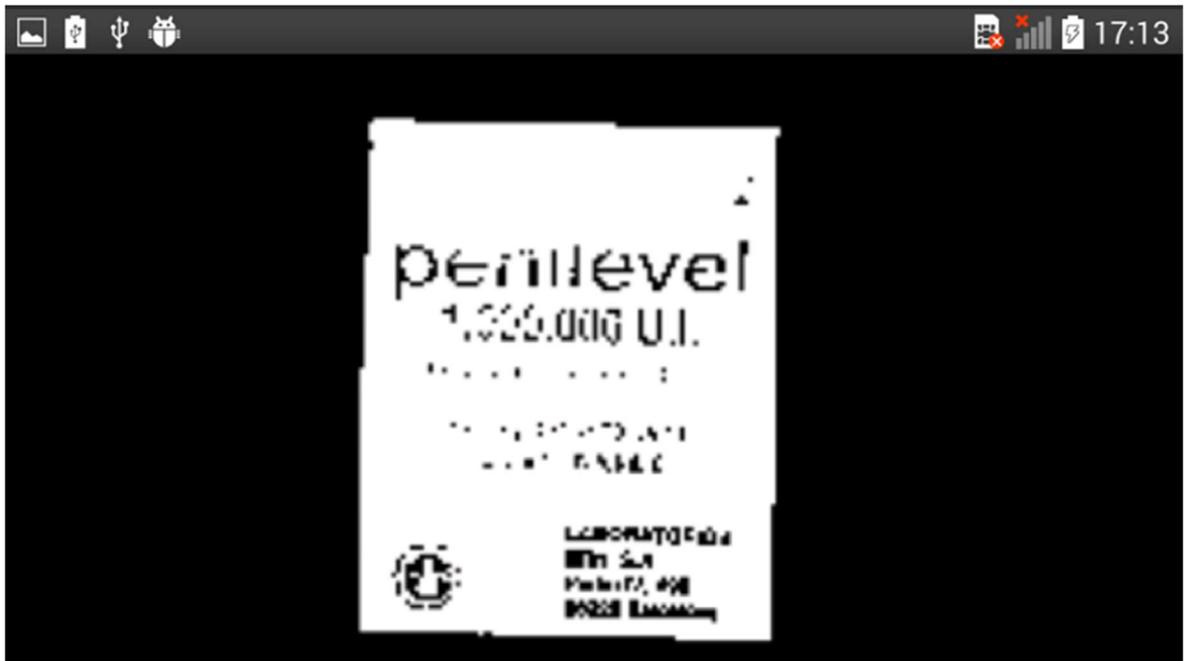
Tras instalar las librerías necesarias para su uso, como con ARToolkit y OpenCV, se iniciaron las pruebas. Las primeras fueron se realizaron para la identificación de textos en inglés, ya que es el diccionario por defecto que usa OCR.

Dependiendo del idioma de las palabras que se requieran tratar se permite usar un archivo llamado diccionario que facilita su identificación. Como nuestro objetivo principal es el de detectar los códigos nacionales de los medicamentos, que son secuencias numéricas, se realizaron pruebas con un diccionario especial orientado a identificar números. Con estas pruebas se vio que este diccionario para números no está muy desarrollado, ya que se identifican mejor con un diccionario para inglés o español. Al final, se decidió usar el diccionario para lengua española, ya que también tenía facilidad al identificar los nombres de los medicamentos, o palabras que aparecen en las cajas de cada uno como *pastillas* o *vía oral*, que podrían servir en el futuro para añadir más información.

5.3 Resultados

En la sección anterior se ha explicado el completo funcionamiento de la realidad aumentada de la aplicación. Ahora se van a mostrar los resultados de ese trabajo de desarrollo.

Para empezar, se va a mostrar el resultado de una de las partes más importantes de nuestra realidad aumentada, la función que realiza la detección de componentes conexas. Durante mucho tiempo de desarrollo y pruebas, la parte de la realidad aumentada tenía el siguiente aspecto:



Con ello se comprobaba que la función detectaba correctamente las componentes conexas. Hay que destacar que la imagen no es directamente la salida que muestra la función, tiene un tratamiento. Como ya se ha explicado, se muestra la segunda componente más grande, ya que la primera suele ser generalmente el fondo de la imagen. Además, se pintaban todas ellas en color negro, salvo la segunda, que se pintaba en blanco para comprobar si era efectivamente la caja de un medicamento.

Tras la detección de muchas cajas y comprobar el funcionamiento de la detección de los códigos nacionales mediante OCR, se pasó a pintar la información de las bases de datos según el medicamento. Por lo tanto, el aspecto final de la navegación por la realidad aumentada tras detectar un medicamento es el siguiente:



Esta imagen es el ejemplo de un medicamento que ha sido registrado previamente por el usuario. De ahí que podamos mostrar el botón de información y las alertas por composición que se muestran tanto para los medicamentos registrados y los no registrados, y además los datos del usuario como son el tiempo hasta la próxima toma y el icono de una pastilla que irá cambiando de color para indicar las pastillas restantes que le queden al usuario en el envase: verde será más de un 66%, cambiará a naranja cuando esté entre un 33% y un 66%, y terminará siendo rojo cuando queden menos de un 33% de pastillas en la caja.

Para los medicamentos de los cuales el usuario no haya registrado su tratamiento se mostrará simplemente las alertas por composición y el botón de información, como en el siguiente ejemplo:



En esta imagen cabe destacar que el medicamento Clamoxyl no tiene en alertas por composición a las personas con fotosensibilidad o a los deportistas por riesgo de *dopping*, si no que se ha forzado a la aplicación a pintar esos iconos cuando detectara el medicamento para mostrar la variedad de alertas, ya que la mayoría de medicamentos tienen como alertas a mujeres embarazadas y lactantes como el Penilevel mostrado más arriba.

6 Conclusiones y trabajo futuro

6.1 Conclusiones

El objetivo principal de la aplicación es el de ayudar a personas mayores con muchos tratamientos de medicamentos que deben seguir día a día y pueden llegar a confundir debido a problemas de su edad.

Para ello se va a usar una de las herramientas más habituales hoy en día, el *smartphone*. Al desarrollar la aplicación para estos dispositivos se facilita la utilización por parte de los usuarios. El proyecto conjunto se basa en un sistema donde el usuario pueda registrar sus medicamentos y sea avisado cuando deba tomárselos, este registro necesita de una foto para que al confirmar que el usuario realiza su toma se compruebe con otra imagen del medicamento tomado que es el correcto. Y la parte más importante correspondiente a este TFG, cualquier usuario con la aplicación podrá apuntar la cámara de su dispositivo móvil a un medicamento y obtener toda la información relacionada con él mediante realidad aumentada.

Merece la pena destacar que se ha buscado desarrollar la aplicación con una interfaz clara, básica y sencilla de cara al tipo de usuario final que la usará. Aunque podrá usarla cualquier persona, nuestro objetivo principal son las personas mayores.

En el ámbito personal, hay que decir que este TFG me ha ayudado a aprender a organizar y desarrollar un gran proyecto conjunto a otras personas, ha sido un proyecto grande en el que no dependía de mí mismo. También me ha servido para mejorar mis aptitudes en desarrollar aplicaciones Android usando nuevas funcionalidades que no se han explicado durante la carrera y además conocer nuevas tecnologías como son OpenCV y OCR.

Finalizar el proyecto me ha dejado unas sensaciones muy buenas, ya que tras todos los problemas que han ido surgiendo se han conseguido ir salvando con la ayuda del resto de integrantes del equipo. Además, desde el primer momento quise realizar un TFG trabajando sobre Android, y este añadía la parte de realidad aumentada que me llamaba muchísimo la atención y no me ha defraudado ya que he encontrado todos los conocimientos adquiridos sobre ella muy interesantes y de cara a mi futuro profesional pueden ser muy útiles, ya que es algo en lo que muchas empresas grandes están invirtiendo recursos.

Por último, decir que me gustaría seguir trabajando y mejorando la aplicación con algunos detalles que no entraban en el ámbito principal del proyecto o no terminaron de realizarse por falta de tiempo. Algunos de ellos se enumeran en el siguiente apartado.

6.2 Trabajo futuro

Este proyecto ofrece muchas mejoras que se pueden implementar en un futuro como son las siguientes:

- **Multicuenta:** Se puede implementar otra base de datos con al cual permitir registrar que distintos usuarios usen la aplicación en el mismo dispositivo. Esto se ha pensado que podría llegar a ser muy útil para residencias u hospitales, con la cual cuidadores de enfermos o ancianos podrían tener una cuenta para cada uno de ellos y recibir distintos avisos.
- **Medicamentos extranjeros:** Una vez realizada la implementación para medicamentos nacionales, se podría ampliar el catálogo de medicamentos que la aplicación identifique con las bases de datos de otros países.
- **Interfaz en otros idiomas:** Con el punto anterior también ayudaría implementar la traducción de la aplicación a otras lenguas.
- **Mejoras en RA:** Actualmente, la navegación por la realidad aumentada muestra las alertas por composición del medicamento tras identificarlo, esto quiere decir que nos muestra cuando no es recomendable para un mujer embarazada o gente con fotosensibilidad, por ejemplo. Además, para los medicamentos que el usuario ha registrado, se muestra también las pastillas restantes y el tiempo hasta la próxima toma. En el futuro, se podrán añadir opciones para que el usuario personalice qué información quiere que muestre la aplicación como el principio activo, la composición o sus efectos secundarios. De cara a personal que trabaje en farmacias, por ejemplo, se podría mostrar el precio del medicamento tras detectarlo.
- **Log:** Se puede añadir funcionalidad a la aplicación, junto a algunos campos en la base de datos para proporcionar información adicional, por ejemplo, al médico, familiar o cuidador de un usuario. Algunos ejemplos de lo que se podría hacer sería obtener un historial de las tomas filtradas por cada medicamento, observar el método de confirmación usado o ver la hora de confirmación de la toma respecto a la hora exacta a la que debería tomarse.
- **Pruebas con usuarios:** Una de las partes más importantes es la de probar la aplicación con usuarios finales que tengan una gran cantidad de tratamientos que tomar al día, para evaluar si la aplicación cumple su principal objetivo, ayudar a estas personas con sus medicamentos.

Referencias

- [1] Statista, <<Android and iOS are the last two Standing>> 2016. [En línea]. Available: <https://www.statista.com/chart/4431/smartphone-operating-system-market-share/>
- [2] Android Developers, <<Dashboard>> 2015-2016. [En línea]. Available: <https://developer.android.com/about/dashboards/index.html?hl=es>
- [3] Wikipedia, <<Scrum (modelo de software)>> 2016. [En línea]. Available: [https://es.wikipedia.org/wiki/Scrum_\(desarrollo_de_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software))
- [4] Wikipedia, <<Realidad Aumentada>> 2016. [En línea]. Available: https://es.wikipedia.org/wiki/Realidad_aumentada#Niveles
- [5] Wikipedia, <<ARToolkit>> 2016. [En línea]. Available: <https://es.wikipedia.org/wiki/ARToolKit>
- [6] OpenCV, <<About OpenCV>> 2016. [En línea]. Available: <http://opencv.org/about.html>
- [7] Wikipedia, <<Reconomiento óptico de caracteres>> 2016. [En línea]. Available: https://es.wikipedia.org/wiki/Reconocimiento_%C3%B3ptico_de_caracteres
- [8] Android Studio, <<Meet Android Studio>> 2016. [En línea]. Available: <https://developer.android.com/studio/intro/index.html>
- [9] Wikipedia, <<Catálogo de especialidades farmacéuticas>> 2015. [En línea]. Available: https://es.wikipedia.org/wiki/Cat%C3%A1logo_de_especialidades_farmac%C3%A9uticas
- [10] Wikipedia, <<Bitbucket>> 2016. [En línea]. Available: <https://es.wikipedia.org/wiki/Bitbucket>
- [11] Wikipedia, <<Trello>> 2016. [En línea]. Available: <https://en.wikipedia.org/wiki/Trello>
- [12] Material Design, <<Introduction>> 2016. [En línea]. Available: <https://material.google.com/>

Anexos

A Manual del programador

Para comenzar a utilizar la aplicación (Medetect), es necesario disponer de los siguientes elementos:

1. Android-Studio
2. URL del repositorio donde se ubica el proyecto Medetect-App

Pasos a seguir para una instalación limpia (desde cero):

1. Acceder a la página web: <http://developer.android.com/intl/es/sdk/index.html>
2. Seleccionar el sistema operativo correspondiente. En el caso de que se instale en un sistema operativo que no sea Windows, se debe seleccionar la opción “Other Download Options” y elegir el sistema operativo deseado.
3. Descargar la última versión indicada de Android Studio
4. Seguir los pasos indicados en la web para configurar correctamente Android Studio. En caso de que la instalación no sea en Windows, seleccionar la opción “Show Instructions for all platforms”
5. Tras la instalación, se deberá seleccionar la opción “Open an existing Android Studio project” que se encuentra en la pantalla inicial de Android Studio (“Welcome to Android Studio”)
6. Seleccionar la carpeta que contiene el proyecto Medetect-App
7. Por último, lo único que hace falta es pasar a la vista de proyecto Android (situado en la parte superior del lateral izquierdo de la pantalla), conectar un dispositivo móvil por USB o utilizar un emulador y ejecutar el proyecto con la flecha de color verde de la barra de herramientas.

B Tabla de tareas de prácticas en empresa

Tabla 4: Tabla de tareas de prácticas en empresa

| Tarea | Horas |
|--|--------------|
| Investigación previa | |
| Reuniones semanales | 20 |
| Familiarización con los nuevos entornos de desarrollo (Android Studio, Datum Free) | 5 |
| Familiarización con las nuevas herramientas (Bitbucket, Git, Trello) | 10 |
| Estudio de pautas y diseño para la aplicación | 10 |
| | |
| Diseño | |
| Diseño de la base de datos | 5 |
| Diseño de maquetas de la aplicación | 10 |
| | |
| Desarrollo | |
| Codificación y desarrollo de la interfaz de la aplicación | 120 |
| Codificación y desarrollo del sistema de notificaciones | 15 |
| Codificación y desarrollo de la base de datos | 10 |
| Desarrollo de la navegación y flujo entre pantallas de la aplicación | 15 |
| Implementación de la base de datos en la aplicación | 20 |
| | |
| Pruebas | |
| Pruebas de la base de datos | 5 |
| Pruebas de usabilidad (interfaz y comprobación de pautas) | 35 |
| Pruebas unitarias | 20 |