

## **Graduado en Ingeniería Informática**

Universidad Politécnica de Madrid

Escuela Técnica Superior de  
Ingenieros Informáticos

### **TRABAJO FIN DE GRADO**

**Desarrollo de una solución nativa y multiplataforma para  
la interacción desde una plataforma móvil con una  
aplicación bancaria**

Autor: Carlos Sánchez Marín

Director: Fernando Pérez Costoya

MADRID, ENERO DE 2019



% La numeración debe ser romana hasta la primera sección

# ÍNDICE

<b>1. Introducción</b>	<b>1</b>
<b>2. Estado del arte</b>	<b>1</b>
2.1. Tecnologías . . . . .	1
2.1.1. iOS . . . . .	2
2.1.2. Android . . . . .	2
2.1.3. Cordova . . . . .	3
<b>3. Otra sección</b>	<b>3</b>
3.1. Una subsección . . . . .	3
3.2. Otra subsección . . . . .	3
<b>4. Resultados y conclusiones</b>	<b>3</b>
4.1. Pruebas . . . . .	3
4.2. Conclusiones . . . . .	3
<b>5. Referencias</b>	<b>3</b>

**ÍNDICE DE FIGURAS**

**ÍNDICE DE TABLAS**

### **Resumen**

Aquí el texto del abstract.

**Palabras clave:** palabra 1, palabra 2, palabra 3...

### **Abstract**

Here goes the abstract text.

**Keywords:** keyword 1, keyword 2, keyword 3...

# 1. INTRODUCCIÓN

En los últimos años el sector bancario ha visto como los pagos con tarjetas ha crecido exponencialmente. Según los datos ofrecidos por el Banco de España, desde 2002 hasta 2017 el número de operaciones creció un 245 % y el importe de las operaciones un 188 %, llegando a un gasto total de 135.246,47 millones de € en el 2017[1].

Este rápido incremento de las operaciones con tarjeta también se ha trasladado al entorno corporativo, generando nuevas necesidades para el control por parte de las empresas de las tarjetas que dan a sus empleados. Con el objetivo de cubrir esta necesidad, tanto Mastercard<sup>1</sup> como Visa<sup>2</sup> han implementado nuevas funcionalidades de control de tarjetas que permiten el control directo por parte de las empresas de sus tarjetas, de manera que se puedan recibir alertas de gasto o bloquear determinados tipos de pago.

Este servicio es ofrecido a través de diversas API REST y en Vipera hemos implementado una aplicación que abstrae a los clientes finales (bancos y empresas) de las API proporcionadas por Visa y Mastercard, ofreciendo una API REST centralizada que simplifica las operaciones. Esta API interactúa con un servicio propietario de Vipera que integra toda la funcionalidad requerida por los distintos clientes finales.

Los SDK desarrollados para la integración de nuestro servicio en las aplicaciones móviles de los clientes pretenden abstraer totalmente de las llamadas al servicio de manera que los desarrolladores tan solo vean funciones totalmente descriptivas sin tener que preocuparse de la configuración necesaria para la interacción. Así mismo, ya que la demo técnica realizada por Vipera está implementada con Ionic<sup>3</sup>, se ha implementado un plugin de Cordova<sup>4</sup> que habilita la llamada a los SDK de iOS y Android.

## 2. ESTADO DEL ARTE

### 2.1. Tecnologías

En el mercado móvil actual iOS y Android lideran con mano de hierro dejando una cuota residual a otros entornos como Windows Phone y otros sistemas menores. En conjunto, iOS y Android acapararon el 99,7 % de las ventas en el primer trimestre de 2018 en España, dejando a Windows fuera de las estadísticas al no lograr el 0,1 % de las ventas totales<sup>5</sup>. Teniendo presentes estos datos, resulta obvio que cualquier tipo de desarrollo fuera de Android e iOS tendría un impacto muy reducido y difícilmente sería rentable para cualquier aplicación que no tenga como objetivo a todos los usuarios del mercado móvil.

---

<sup>1</sup><https://developer.mastercard.com/product/spend-controls>

<sup>2</sup><https://developer.visa.com/capabilities/vctc>

<sup>3</sup><https://ionicframework.com/>

<sup>4</sup><https://cordova.apache.org/>

<sup>5</sup><https://bit.ly/2Cu8NVy>

### 2.1.1. iOS

El sistema operativo iOS es un sistema propietario desarrollado por Apple<sup>6</sup> para sus dispositivos móviles iPhone, iPad y iPod Touch. Fue presentado junto con el primer iPhone en junio de 2007. En un principio Apple no iba a proporcionar SDK para desarrollo de aplicaciones nativas por parte de terceros, pero ante la reacción negativa de todos los desarrolladores, Apple reconsideró su decisión y terminó lanzando la primera versión del SDK en Marzo de 2008 junto a la segunda versión de iOS.

El SDK originalmente proporcionaba un acceso muy limitado a las funciones del sistema operativo, además de tan solo proporcionar soporte para el lenguaje Objective-C<sup>7</sup>. Año tras año Apple ha actualizado su sistema operativo móvil junto con el SDK aprovechando el lanzamiento de los nuevos iPhone.

En 2014, Apple lanzó Swift<sup>8</sup>, un nuevo lenguaje más moderno que sustituyó a Objective-C como lenguaje de referencia para el desarrollo de aplicaciones iOS. No obstante, ambos lenguajes son actualmente soportados por el SDK de Apple, aunque la documentación tan solo es actualizada para Swift.

### 2.1.2. Android

Android<sup>9</sup> fue creado en 2003 como un fork del kernel de Linux para dispositivos móviles. Inicialmente fue concebido como un sistema operativo móvil inteligente que estuviese pendiente de la localización del usuario y sus preferencias. En 2005 la empresa propietaria del sistema operativo fue comprada por Google, que se hizo cargo del desarrollo. Dos años más tarde, Google hizo público el primer SDK en fase beta para Android, que fue distribuido a desarrolladores y fabricantes de dispositivos móviles. No fue hasta septiembre de 2008 que se lanzó al mercado el primer dispositivo con Android.

Las primeras versiones del SDK proporcionado por Google contaban tan solo con soporte para el lenguaje Java<sup>10</sup>. Al contrario de su principal competidor, el código fuente de Android siempre ha sido open source<sup>11</sup> lo que ha atraído a una gran cantidad de desarrolladores y fabricantes. De todos modos, la práctica totalidad de los fabricantes integran en sus terminales la versión de Android propietaria de Google, que incluye integración con las apps de Google bajo la capa Google Play Services<sup>12</sup>.

En el evento Google I/O de 2017, Google anunció de manera oficial que se añadía un nuevo lenguaje soportado al SDK de Android<sup>13</sup>, Kotlin<sup>14</sup>. Este lenguaje desarrollado por la empresa JetBrains<sup>15</sup> funciona sobre la Java Virtual Machine (JVM) de manera que

---

<sup>6</sup><https://apple.com/es/>

<sup>7</sup><https://developer.apple.com/library/archive/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html>

<sup>8</sup><https://swift.org/>

<sup>9</sup><https://www.android.com>

<sup>10</sup><https://www.java.com>

<sup>11</sup><https://source.android.com/>

<sup>12</sup><https://developers.google.com/android/guides/overview>

<sup>13</sup><https://blog.jetbrains.com/kotlin/2017/05/kotlin-on-android-now-official/>

<sup>14</sup><https://kotlinlang.org/>

<sup>15</sup><https://www.jetbrains.com/>

es compatible con Java, por lo que es totalmente compatible con el código Java y puede coexistir con este en una misma aplicación.

### **2.1.3. Cordova**

En el año 2011, Adobe Systems<sup>16</sup> compró la empresa Nitobi y renombró su producto como Phonegap<sup>17</sup>. Así mismo, lanzó una versión open source del producto nombrada Cordova que es mantenida por la Apache Software Foundation<sup>18</sup>. Este framework permite el desarrollo de aplicaciones híbridas basadas en la web, embebiendo una aplicación web dentro de una aplicación nativa para el sistema operativo deseado. Esta solución se acerca a la idea que tenía Apple de aplicaciones móviles en un principio, pero va más allá otorgando un sistema de plugins nativos que permiten la interacción con el sistema operativo. Los plugins proporcionan una implementación en código nativo (Objective-C para iOS y Java para Android) que será invocada por un wrapper escrito en Javascript<sup>19</sup>. En tiempo de ejecución Cordova es capaz de saber sobre qué plataforma está ejecutándose e invocar al código nativo de esta, con lo que abstrae a los desarrolladores de aplicaciones de esta gestión y facilita la escritura de código.

## **3. OTRA SECCIÓN**

### **3.1. Una subsección**

### **3.2. Otra subsección**

## **4. RESULTADOS Y CONCLUSIONES**

### **4.1. Pruebas**

### **4.2. Conclusiones**

## **5. REFERENCIAS**

- [1] B. de España. (2018). Estadísticas Tarjetas para la Web 2T-2018, dirección: <https://www.bde.es/f/webbde/SPA/sispago/ficheros/es/estadisticas.pdf> (visitado 05-11-2018).

---

<sup>16</sup><https://www.adobe.com/>

<sup>17</sup><https://phonegap.com/>

<sup>18</sup><https://www.apache.org/>

<sup>19</sup><https://developer.mozilla.org/bm/docs/Web/JavaScript>