Curriculum Vitae

Erik Medina

**Worldpay** (Android Developer)

*(Feb 2019 - Present)*

We can categorize the Worldpay’s projects into 2 groups. The former is in charge of the Android/iOS app and the latter is in charge of the SDK which allows to make payments with payment cards. This SDK is integrated into the Worldpay app and into the customers’ apps (Tesco, Boots, M&M, etc.). Currently the SDK only supports **one** payment device (a the device where you insert or tap your card).

Worldpay has 2 objectives:

* Make the SDK support multiple payment devices (so they can sell the SDK to more merchants)
* Convert a smartphone into a device able to **accept** contactless payments (i.e. you can make payments tapping a contactless card on the smartphone)

I work in the SDK group. In this team we are iOS devs, Android devs (4 devs), UX/UI designers and testers. Focusing in the Android devs, we **all** work in the SDK, in the wrapper that makes the SDK to support multiple devices and in the project that converts the smartphone into a payment device. Some of our duties are:

* Improve software architecture (clean code, MVVM pattern with LiveData and ViewModel, remove unused/deprecated libraries, apply SOLID principles, etc.)
* Implement features and fix bugs
* Implement Unit Tests using JUnit, Hamcrest, Mockito and MockK (to support Kotlin features such as Extension functions)
* Review other developers’ code through Pull Requests
* Propose improvements (e.g. I suggested using Gitflow for the control of the branches)
* Because converting a smartphone into a payment device is a risk, we have security concerns and we have to use tools in order to trigger security events when the smartphone is compromised (events such as erase app data, kill the app if a debugger is detected, advise user about the risk, advise the server about the security events, etc.)
* Migrate from Java to Kotlin

**Oraclize** (Android Developer)

*(Sep 2018 - Jan 2019)*

* Update the Android SDK of the application to the latest version (currently API 28)
* Update the integration with the SafetyNet API every time there is a new release of it. We use this API in the application to verify that the device is not compromised
* Implement new features, e.g., now the application is able to perform **Ethereum** transactions. We had to integrate the library https://web3j.io/ for this purpose.
* Implement Unit test to ensure that the application works as expected. For this, we use tools such as JUnit, Roboelectric and Mockito.
* Code reviews, my coworker and myself open Pull Requests to check the code before merging in the main branch

**O2O** (Android Developer)

*(Mar 2018 - Jun 2018)*

* Start projects from scratch: split project requirements in development tasks, estimate hours of these tasks, think about the tech stack for the projects and implement architecture/features
* Maintenance/update of existing projects: implement new features, fixing bugs and new release in the Google Play
* Agile methodology
* Show advances to the client

**Telefónica** (Android TV Developer)

*(May 2017 - Feb 2018)*

* Development of a greenfield project in Kotlin. This was an hybrid application.
* Implement features and integrate APIs
* Implement design delivered by the design team
* Fixing bugs that the QA team reported
* Agile methodology

**SDOS** (Android Developer)

*(Mar 2017 - Apr 2017)*

* Maintenance of existing applications: implement new features and fixing bugs

**Vector** (Android Developer)

*(Apr 2016 - Feb 2017)*

* Greenfield projects and existing applications for different clients
* Implement features, integrate APIs/SDKs
* Brainstorming meetings
* Agile methodology
* Show demos to the client

**Weplan** (Android Internship)

*(Nov 2015 - Feb 2016)*

* Data and voice usage application: Weplan
* Development of new features and fixing bugs
* Testing of app and checking/updating the database
* Teck Stack: Test Automation (Appium)