Adam Świderski

Android Developer

Poznań, Poland

+48 505304977

http://asvid.github.io/

http://github.com/asvid

adam.swiderski89@gmail.com

About me

I'm a clean code fan with constant knowledge hunger. I've started career as a frontend developer, but I've switched to Android as soon as it was possible. I like to use new technologies and approaches in projects, but I'm far from hype-driven development.

Lately I'm trying to step outside Android world and work a bit on backend and iOS.

I'm mostly focused on mobile apps, but I'd really want to make steps in game development.

Skills

Tech

Java, Kotlin, Android SDK, Realm, Room, RxJava2, Dagger2, JUnit, Spock, Espresso, Gradle, JavaScript, NodeJS, Python

Tools

Android Studio, IntelliJ, Firebase, Git, Jira, Bamboo, Jenkins, Dokka, SonarQube, Adobe Photoshop

Practises

Unit testing, design patterns, Continues Integration/Delivery, Scrum, Prince2

Education

2014-2016 **Computer Science Msc** — Poznań University of Technology
Software Development Technologies - master thesis "Home inventory management system"

2010-2013 **Computer Science undergraduate studies** — WSNHiD
Internet Technologies

Courses

2014 **PRINCE 2 Foundation** — Altkom
AXELOS Global Best Practice, Licence number 02807643-01-EZYL

2013 **Graphic Design** — WSNHiD
Adobe Photoshop, InDesign, Illustrator

Experience

2019 **Payworks** — Android Developer
The world's best Point of Sale solutions

2015-2019 **Fibaro (Fibar Group)** — Android Developer
Manufacturer of wireless home intelligence system, available in over 100 countries and in some being synonymous of home intelligence.

2013-2015 **Lobo Group** — Frontend / Android Developer
Interactive agency focused around e-learning platforms.

Projects

Fibaro Android App

Smartphone and tablet app to control intelligent home system. Communication based on Volley and GCM. It's an old app, refactored over time, by using MVP pattern and unit tests in in Spock / Groovy. For videocalls we used Linphone Library. We also provided widgets and Android Watch App. Recently, I managed to establish and introduce a strict codestyle and static code analysis with SonarQube. App releases are automated with Fastlane. Also unit testing, UI testing and documentation generation are automated with Bamboo CI server.

Dinegra

It's a car assistant Android app. User can register fuel usage, call help, see POI on Google Map. App tracks drivers routes, that can be used in challenges or events (like Endomondo). OpenGL is used to create animated bot, that can talk with you. App also can be controlled by voice commands. App can also be a silent thief alarm - when armed, it will send you emails where the car is if it moves from parking place without disarming. We also made Websocket voice operated chat for drivers nearby, like CB radio.

Frigo

For my graduation project I used Kotlin as weapon of choice for Android app. It uses Realm for storage and Retrofit for communication. App uses GCM for instant data synchronisation and request caching if internet is not available. Backend of app is made in ExpressJS working on Heroku with MongoDB database. During development, I started an open source library for Android notifications called - Notti.

Counter

Playground app that contains resizable widgets, charts made in MPAndroidChart, Relam database and Shared Element Transition.

Interests

- Good (or really bad) movies
- Music, playing on guitar
- Making world a better place - one line of code at the time

GdzieTaBiedra

App is showing shops on map with their opening time info and can turn navigation to selected one. Realm was used for storage and Retrofit for getting data from server. App was build around Uber RIBs design approach.

AirRide

Hobby project, pneumatic car suspension controller. Android app is connected via Bluetooth with Arduino module controlling the air valves. App allowed controlling suspension manually with pressing buttons, using accelerometer, and creating sequences stored in Realm. Communication was wrapped in Command Pattern which made changing the API approach a lot easer.