# Summary

I’ve been coding on iOS apps for the last three years although I have built Android apps as well in the past. I've been passionate about technology for as long as I can remember. I wrote my first commercial lines of code 10 years ago as a freelance web developer turning to mobile apps development along the way.

I am a practitioner of user-centric product development, I believe that running lean and agile is the way to create value for your users.

# Frameworks and tools

- AVFoundation, CoreAudio, CoreAnimation, CoreData, UIKit, MapKit, Security

- Git, Xcode, Instruments, AppCode, Fastlane, Jenkins, Bots, Charles proxy, mitmproxy, Wireshark, Paw, Zeplin, Sketch, PaintCode, Dash

# Employment history

** **

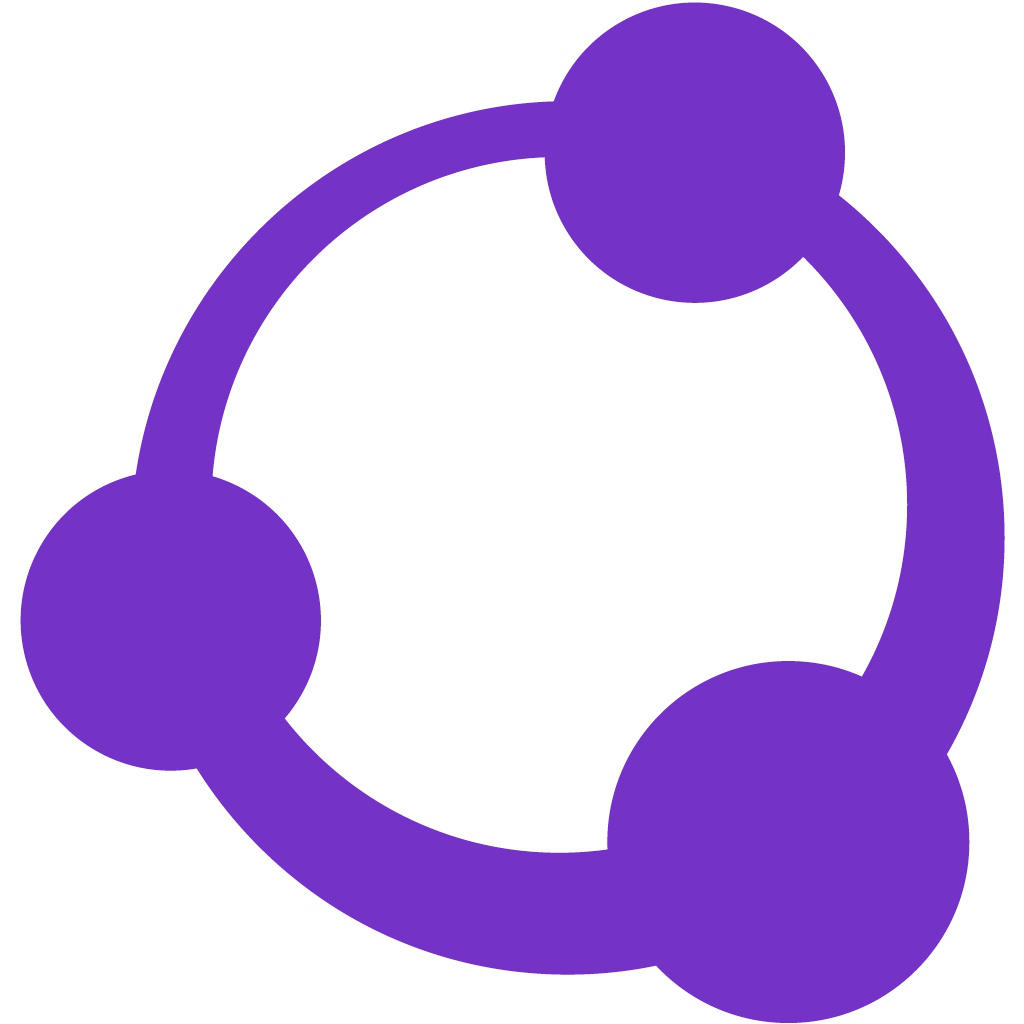
**iOS Developer, London, UK** *December 2014 - March 2016*

Composed is a classical music streaming app, a startup venture from Universal Music Group. Until the summer of 2015, it was hosted and managed by Made by Many, a product innovation studio based in London and New York, before integration back into Universal Music Group.  
I was in charge of developing and maintaining the iOS app. During my time there I’ve increased unit test coverage by 50%, replaced the audio engine in the app with an open-source implementation, added new UI in universal storyboards with auto-layout and written all new code in Swift.  
Additionally, at Composed we worked as a cross functional team and were all also involved in product-related tasks such as analytics review, user interviews, product steering sessions or feature sketch sessions.

**      **

**iOS Developer, Cluj-Napoca, Romania** *September 2012 - July 2014*

I was mainly involved in developing native apps for iPhone and iPad. I also performed code reviews for Audi AG, analyzing code bases of more than 20 internal and consumer iOS apps. Additionally I offered to help out on Android projects when needed.  
Here are some apps that I build from scratch:   
- Tanke: a location based mobile app that helps electric car owners in Vienna find charging stations around them.   
- mobiLEOS: product targeting the health sector in Germany. It helps health service providers manage appointments with patients and deal with paperwork involving treatment and payment.   
- De-Touro: product operating in the German health market. It enables cab drivers to bid on patient transport jobs offered by health insurance companies using their mobile devices.

**Small Footprint, Inc**.   

**Mobile App Developer, Cluj-Napoca, Romania** *November 2011 - August 2012*

Started out working on an app built with Titanium API but realized this technology was becoming a performance bottleneck. Then switched to native iOS development, learning along the way, I built two enterprise apps, one supporting iPhone and the other for iPad. My last project at this company was developing a simple 2D puzzle game for Android phones and tablets (Java) from scratch.

**OSF Global Services**

**JavaScript Developer, Cluj-Napoca, Romania** *October 2010 - August 2011*

I worked with Javascript both on the server-side and inside the browser (jQuery) while integrating custom e-commerce solutions based on the Demandware platform. I also worked on developing a few mobile applications for Android and Blackberry using the native APIs (Java) and Appcelerator’s Titanium API (Javascript).

**OgreCore.com - self employed**

**Web Developer, Cluj-Napoca, Romania** *April 2005 - August 2010*

Development of web applications using PHP/MySQL/HTML. Developed e-commerce websites, CMS, Wordpress customization, various front-end work.

# Education

**Bachelor’s degree – Electronics, Telecommunications and IT** *2006 – 2010*

**Technical University of Cluj Napoca**

Wrote my bachelor’s thesis at Vrije Universiteit Brussel as part of an Erasmus Programme. My project involved developing a sound-based localization method for determining the position of wireless sensor nodes. I developed an algorithm that would determine the position of a wireless sensor in a cartesian system of coordinates by using sound via on-board speakers and microphone. Wireless sensor nodes were running TinyOS, a real-time operating system and the code was written in nesC.

# Interests

- sports: basketball, tennis, golf, table-tennis

- I enjoy contributing to open source projects

- reading about computer science, software business, startups and design