

Back in 2013 during my Bachelor studies, I started learning Java and Android app development as a hobby and after spending a whole summer, managed to make a simple tool box application. After a while, I converted the codebase to Kotlin and published it to the market. That was my first enjoyable experience of how satisfying it is to develop and publish android apps. Moreover, how much easier it is to write/read code in Kotlin (especially compared to the boring and verbose Java!).

After finishing my master of Computer Science, I started as an Android developer intern in [KAR Global](#) company and over the course of 4 months, I worked on [TradeRev Android app](#). Some of my responsibilities were:

- Cooperating in migration from Android View library to Jetpack Compose.
- Migration from LiveData to SharedFlow/StateFlow API.
- Adding functionality for uploading photos from gallery to the server.
- Further developing the dependency injection module of the app using Koin library.
- Migrating DataSources of the app from PositionalDataSource to PagingSource (from paging 2 to paging 3)
- Migrating legacy code's architecture from MVP to MVVM.
- Migration from ButterKnife/Synthetic to ViewBinding library.
- Migrating some legacy Java code to Kotlin.
- Resolving business logic issues/bugs.

After finishing my work contract, I'm currently working on some personal projects; the most notable of them is named [Game Hub](#). It's a native Android app for browsing the latest news about vidoe games; published to Google Play store. Some of the technologies I used in this app are as follows:

- Jetpack Compose + Accompanist for UI
- Lottie animations in Compose
- Dagger hilt for DI
- Room + DataStore for data storage
- okHttp + Retrofit for network connection
- JUnit + Turbine + Truth for unit testing
- MockK for mock testing
- DownloadManager
- MVVM + Clean architecture

Kotlin programming language (with all of its sub projects) is currently my favorite technology and I hope that my next job pushes me towards digging deeper in this realm of the tech world; be it Kotlin JVM, Kotlin Multiplatform Mobile, or Kotlin JS.