

# Julian Rosser

Brixton, London | 07403664394 | [julianrosser91@gmail.com](mailto:julianrosser91@gmail.com) | [github.com/Julesssss](https://github.com/Julesssss)

Motivated developer with four years experience building modern Android applications, focused on Kotlin, MVVM, and RxJava and Dagger.  
Current areas of interest include state management, architectural design, and Reactive programming with Rx.

## PROFESSIONAL EXPERIENCE

---

### Senior Android Developer (Contract)

*Feb 2019 — Present  
London*

#### Verv Energy

##### Verv Home Energy Assistant - [Play Store](#)

- Introduced reverse tunnel networking solution which enabled out of local network connection to consumer hub.
- Took technical ownership role for two large cross-platform features, taking on project owner responsibilities.
- Used Strategy Pattern and custom data structure for switching live graph data sources and styles seamlessly at run time.
- Introduced Continuous Integration (Docker/Jenkins), including unit test suite, build deployment, KTLint and docs generation.

### Android Developer

*Apr 2017 — Feb 2019  
London*

#### Ostmodern

##### Arsenal (Arsenal F.C.) - [Play Store](#)

- Architected a modern Android project consisting of Kotlin, MVVM, Data Binding, RxJava and Dagger.
- Responsible for Android application, leading the project through agile methodology and mentored a junior developer.
- Solved client requirement for an offline-first architecture using Observer Pattern, Repository Pattern, RxJava data flows and Realm.
- Ensured codebase quality with unit/automation test coverage using Junit, Mockito and Espresso.
- Set up Continuous Integration pipeline including Ktlint, Kotlin doc generation, Fabric for automated release process.

##### F1TV (Formula 1)

- Architected a modularised OTT streaming app in Kotlin with 5 integrations; F1 for live race data via websocket, 21 video streams from NBC, CSG for authentication, GooglePlay for subscriptions and Skylark for dynamic CMS content.
- Synchronised multiple concurrent live video streams and race data every 100ms, handling delay, entitlement checks and recalibration with minimal user impact. Heavy use of Rx for managing data flow with Flowables and Publishers.
- Created authentication and GooglePlay in-app purchase library, which was consumed by third party app via Maven.
- Built Android/Amazon TV module which shared ViewModels, business logic and data package with the mobile app.

##### BBC Player (BBC Worldwide)

- Proposed MVP architecture improvements, leading to more performant and faster networking.
- Migrated project to Kotlin, starting with data objects, eventually using Kotlin for every additional feature or fix.

### Android Developer / Consultant

*Feb 2016 — Apr 2017  
London*

#### Reply

##### CinemApp (Brandiment) - [Play Store](#) - [Case Study](#)

- Built a cross-platform, location-aware social app for cinema showings using Firebase as a backend.
- Built native group messaging solution, push notifications, video playback, friend system, and social sign in.

##### Delhaize and AB Click2Shop (Delhaize Group) - [Delhaize Play Store](#) - [AB Play Store](#)

- Maintained two of the largest European supermarket apps in a single code base, partitioned with Gradle configurations.
- Pitched addition of Material Design and new SDK components to the client, leading to additional client spend.

## PERSONAL PROJECTS

---

### NASA Astronomy Photo of the Day - [GitHub](#)

*Jan 2019*

- Demonstrating a modern architecture consisting of Kotlin, MVVM, RxJava, Koin, LiveData and lifecycle aware ViewModels.
- View state is represented by an immutable data class emitted as a LiveData subscription from the ViewModels.

### Android MVVM - [GitHub](#)

*Jun 2018*

- Created a simple wage change calculator in Kotlin to demonstrate the MVVM architecture pattern with data binding.
- Used TDD process for faster development cycle, simplified debugging and high test coverage.

Designed to have most pleasing and intuitive UI in its category on the market.

- Solved challenges including scheduling notifications months in advance and communicating between services.
- Migrated live user databases to Firebase remote storage. Enabled Social sign in, data backup and other user requests.
- Handled user support through email and implemented features and bug fixes based on feedback.

**Alauda** - [GitHub](#)

Apr 2017

- Built to demonstrate the MPV pattern, in conjunction with Realm data persistence. Used presenter layer to maintain state through rotation.

**TECHNICAL COMPETENCIES**

---

- **Android:** Kotlin, Java, Gradle, RxJava, Live Data, Dagger2, Koin, Retrofit, Realm, Room, Live Data, Extension Functions, Sealed Classes.
- **Architecture Patterns:** MVVM, MVP, Clean Code, Data-binding, Lifecycle aware view-models, Observer Pattern.
- **Integrations:** Firebase Database, Firebase Remote Config, Analytics, AWS Lambda, Fabric, Jenkins, Urban Airship.
- **Testing:** JUnit, Espresso, Mockito, TDD.
- **Productivity:** Jenkins task configuration, Git flow, GitHub, BitBucket, Dokka, Jira, Confluence, Agile, Scrum, Target Process, Slack

**PUBLICATIONS**

---

[Android Data Binding: A practical experience](#)

Mar, 2019

[Handling Gradle dependencies in multiple module Android projects](#)

Oct, 2018

[A modern guide to adding Dagger to a Kotlin Android app](#)

July, 2018

[Auto generating documentation for Android Kotlin](#)

June 2018

**COURSES**

---

[Developing Android Apps](#) (Google)

2015

[Programming Mobile Applications for Android Systems](#) (Maryland University)

2014

[Introduction to Interactive Programming in Python](#) (Rice University)

2013