

# Android App Developer: Criteria

Raileasy are looking for an experienced Android developer to work with us on enhancing our mobile application, introducing innovative features and improving the user experience to make our app the go-to option for booking UK rail tickets.

You will work collaboratively with our existing in-house iOS developer, as well as the wider engineering team, to help push forward our mobile offering. Where possible, we hope to be able to provide feature parity for the two platforms, so an ability to work collaboratively on shipping new features is important.

## Essential

- Experience with writing Kotlin, Java and developing for Android\*
- Passion for great UX, and an ability to implement beautiful, functional user interfaces
- A solid understanding of the Google Play Store, Google's policies and the processes involved in publishing releases
- Ability to independently learn an existing codebase, troubleshoot issues in code you may not be familiar with and reduce technical debt as appropriate
- Familiarity with writing technical documentation and writing automated tests (JUnit/Espresso)
- Willingness to work remotely with occasional travel to London, Leamington Spa or other mutually convenient locations (for e.g. hack days to collaborate in-person)
- Ability to take ownership of day-to-day Android app development, whilst working in a small engineering team whose other members will mostly be focused on backend, iOS and web development
- Understanding of Git

\* This doesn't necessarily need to be professional experience.

## Desirable

- An interest in public transport/rail
- Experience with the Jetpack Compose UI toolkit
- Experience maintaining CI(/CD) pipeline configurations (e.g. CircleCI, Travis, GitLab, Bitbucket Pipelines, GitHub Actions etc)

Raileasy engineering culture:

- Open/transparent, with the ability to shape company direction & have your voice heard
- We'll give you autonomy and respect your skillset
- Small, close-knit engineering team with a willingness to wear different hats, collaborate and support each other
- Flexible working hours

# Android App Developer: Q&A

## What does our app do?

The TrainSplit app allows customers to purchase train tickets and save money with ticket splitting. The app is written predominantly in Kotlin (81% by lines of code) and our minSdkVersion is 26. Customers can authenticate, perform journey searches, review search results, purchase tickets (we use Braintree's drop-in Android SDK for payments), download expense receipts and view their eTickets and upcoming journeys in-app.

## Why do we want to bring an Android developer on board?

Our app was originally developed by an agency who we have worked extensively with in the past. We're happy with the work that they have produced for our mobile applications, but want to strengthen our own in-house development capabilities so that we have full control over feature development, release schedules, technical debt etc.

We previously took this decision with our iOS app, and have been able to build up in-house expertise, leading to being able to ship feature improvements (a new journey planner interface, better visibility of disrupted journeys, improved Apple Wallet integration, a new E-Tickets screen and more). This has validated our decision to invest more and further expand our in-house team.

We've been able to make a number of minor improvements to the Android app (visibility of disrupted journeys, ability to recover 'lost' bookings made logged-out, accessibility fixes for visually impaired customers, security improvements, dependency updates), we've improved our continuous integration workflows and have made a number of releases on the Play store - so whilst we anticipate this role to be one with overall responsibility for our app, you would benefit from the existing understanding of the codebase within the team.

## What does the app do to book tickets?

We have our own Raileasy HTTP API (documented with OpenAPI spec v3, docs specified w/ Swagger) which supports both JSON & XML requests/responses. The Android app communicates with the API using OkHttp to issue JSON requests/responses, and (de)serialises data using Gson.

As part of the ticket booking process, we need to have some reference data stored locally (e.g. list of stations, railcards, train operating companies). We persist these, as well as some customer information that we need available offline, using the *Room* persistence library and an underlying SQLite database.

## What can you tell me about the app's architecture?

The app broadly follows the MVVM design pattern, and makes extensive use of interfaces for e.g. data repositories, so that we can easily write mock/fake implementations for use in automated tests that don't touch the API. Concrete dependencies are retrieved using the service locator pattern. We use Firebase for authentication, crash logging and analytics.

## How is the app tested?

There is a reasonably extensive collection of "small" tests in the project (392 test methods across 33 test classes) to test business logic with mock data. We also have a number of Espresso tests to ensure components of the app come together to produce a UI that responds correctly to user interactions, using stubbed data.

For manual testing, we have an API staging environment which allows booking tickets with test card details and generating test e-Tickets.

## How would I be working in this role?

We would look to set-up prioritisation meetings for Android to collaboratively determine where development effort should be concentrated. These meetings would be with technical staff, informed by a steer from the business. We'd like to encourage a regular release schedule that enables a small number of features/bug fixes to be focussed on and released on a continuous basis. We have a backlog of ideas that we think would be valuable and many of these tickets involve improving feature parity with our iOS app and our retail web application (at <https://trainsplit.com>). Additionally, we'd want the successful candidate to be interested in emerging features announced at e.g. Google I/O with an eye to potentially making use of interesting new Android SDK features to improve our user experience.

Code changes across all of our projects are generally proposed as pull requests and reviewed by members of the team. Merging requires various checks to pass, and we run all tests for every commit. Release builds and signing are automated. Familiarity with a Git repository hosting/collaboration site such as Bitbucket, Sourcehut, Gitlab, GitHub etc would be useful.

We'd be delighted to recruit someone with an interest in shaping the design of our API, too. We think the developer experience is important and we are happy to make changes to requests/responses/API flows to better support our mobile apps. The Raileasy API is used outside of the TrainSplit apps too, so any improvements to the endpoints we offer or the documentation will also help our external API clients.

## What if I'm unfamiliar with rail data / journey planning / fares?

Full training would be provided on rail retailing domain matters. An interest in/enthusiasm for public transport / rail would be desirable for this role but is not essential. We like to think that you might develop an interest if you don't already have one at the point of applying for the job!