

Description

Intended User

Features

User Interface Mocks

Main Screen

Detail Screen

Repository and Contributor Item

Key Considerations

How will your app handle data persistence?

Describe any corner cases in the UX.

Describe any libraries you'll be using and share your reasoning for including them.

Describe how you will implement Google Play Services.

Next Steps: Required Tasks

Task 1: Project Setup

Task 2: Implement UI for Each Activity and Fragment

Task 3: Your Next Task

Task 4: Your Next Task

Task 5: Your Next Task

GitHub Username: davidllorca

Trending Repos

Description

The main purpose of Trending Repos is browsing the most famous source of repositories platform. You can search the most popular projects by name or programming language and add projects to favorites list also.

Intended User

Who wants a quick access to Github and its popular repositories.

Features

- View trending repos
- Search repos by name or programming language.
- Bookmark repos.

User Interface Mocks

Main Screen



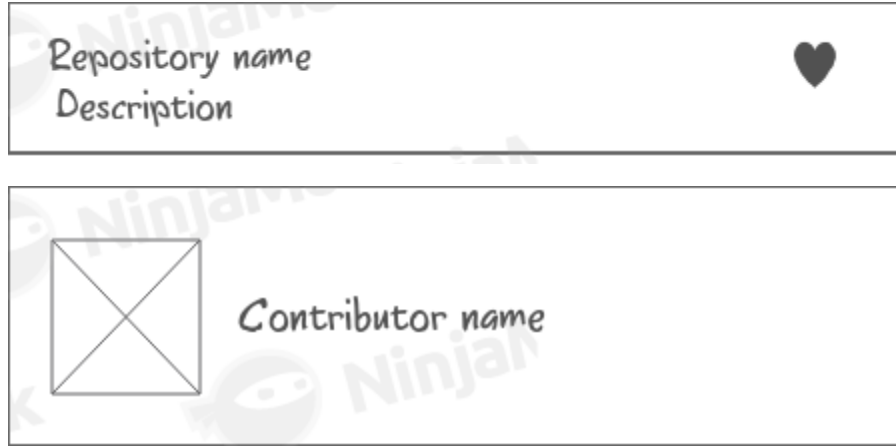
- List trending repositories.
- Search by specific term.
- List favourites repositories.

Detail Screen



- Show details of target repository.
- Add to favourite list.

Repository and Contributor Item



Key Considerations

How will your app handle data persistence?

App use Room library to persist favorites repositories.

Describe any edge or corner cases in the UX.

To avoid large call's responses there will be a cache that saves the searches along the user session.

Describe any libraries you'll be using and share your reasoning for including them.

Picasso to handle the loading and caching of images.

Architecture Components Room to persist local data.

Architecture Components ViewModel to implement a MVVM architecture.

Retrofit 2 to make remote calls.

RxJava 2 to handle threading.

Android Data Binding to avoid boilerplate declaring the UI.

Dagger 2 to provide the dependencies across the app.

Describe how you will implement Google Play Services or other external services.

Firebase Analytics to track which are the most searched terms.

Crashlytics to report app's crashes.

Next Steps: Required Tasks

Task 1: Project Setup

- Configure all libraries used.

Task 2: Implement data layer

- Create DAO to manage local persistence.
- Create model for Repository and Contributor

Task 3: Create remote model response

- Create model for Github call's responses.

Task 4: Implement local API (Repository pattern)

- Create accessors to provide data.

Task 5: Implement remote calls to Github API

- Implement local API to connect with remote server.

Task 6: Implement UI screens

- Implement List layout.
- Implement Detail layout.

Task 7: Implement TrendingReposViewModel

- Create ViewModel with logic to list the repos and other search options.

Task 8: Implement DetailRepoViewModel

- Create ViewModel with logic to display detail information of target repository.

Task 9: Implement Save Repo feature locally

- Implement logic to save a favorite repo.

Task 10: Implement SearchRepoViewModel

- Create ViewModel with logic to list the repos by search options.

Task 11: Create proper Activities

- Implement MainActivity and bind its viewmodels properly.
- Implement DetailActivity and bind its viewmodels properly.