Kutumb Android Assignment

Goal

The goal of this assignment is to build a simple app which shows the current trending Github repositories fetched from a public API. The design and business specifications have been provided which we expect you to follow as closely as possible.

We have deliberately kept the app simple enough for everyone to attempt it, but we are keen to see the approach you take to solve it. You have the freedom to give your best into it and demonstrate your skills for us to evaluate you better.

Requirements

- The app should fetch the trending repositories from the provided public API and display it to the users (refer fig. 1).

- While the data is being fetched, the app should show a loading state (refer fig. 2). Shimmer animation is optional.

- If the app is not able to fetch the data, then it should show an error state to the user with an option to retry again (refer fig. 6).

- All the items in the list should be in their collapsed state by default and can be expanded on being tapped (refer fig. 4).

- Tapping any item will expand it to show more details and collapse any already expanded item. Tapping the same item in expanded state should collapse it.

- The user should have an option to persist the expanded state of a card for subsequent launches.

- The app should be able to handle configuration changes (like rotation).

- The app should give a pull-to-refresh option to the user to force fetch data from remote (refer fig. 5).

- **Bonus:** assign repository language as header in the list for each group in the repositories list sharing the same language (refer fig. 7).

- **Brownie Points:** the user can have an option to star a repository and can access the starred repositories from a separate screen.

- **Brownie Points:** The app can have offline support. Once the data is fetched successfully from remote, it should be stored locally and served from cache thereafter till the cache is not expired.

- Your assignment may be prioritised if it follows granular & meaningful Git commit history (refer here), is based on a well structured architecture, and if the code is easily comprehensible and readable.

API details

- The complete API docs are available <u>here</u>. You need to fetch the data from <u>/repositories</u> endpoint.
  You will get a list of the trending repositories as a response from this API.
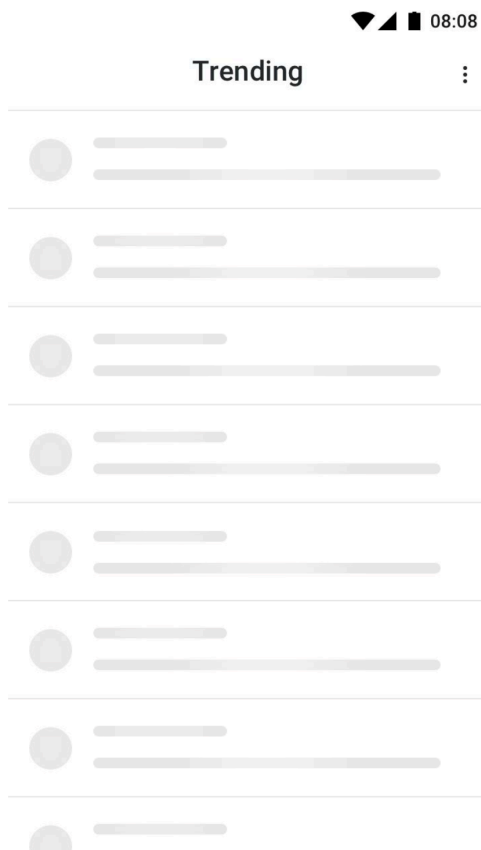
Design Details
Below are the designs all the screens and components of the app.

Given the size of this project, a span of 1-2 days should be enough for you to comfortably complete it. With that aside, we would be interested to see your submission beyond that span as well.
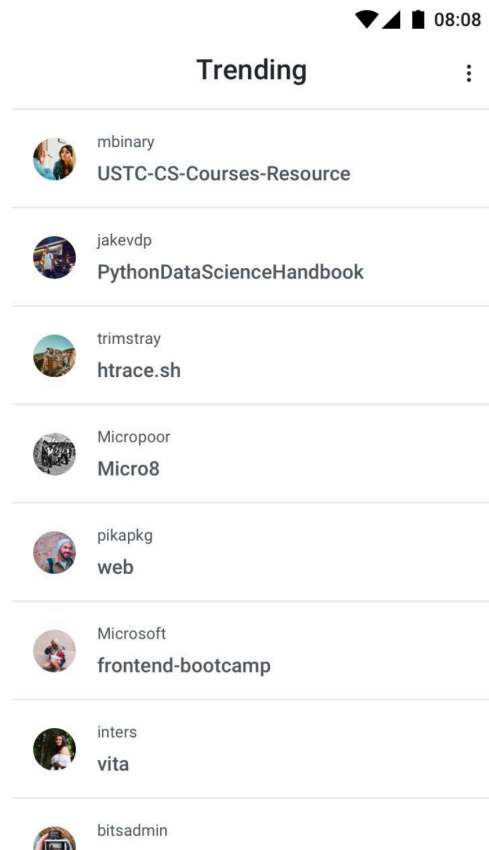
Submission Guidelines

Once you have completed the assignment, upload it to GitHub with an arbitrary name and no description or images.
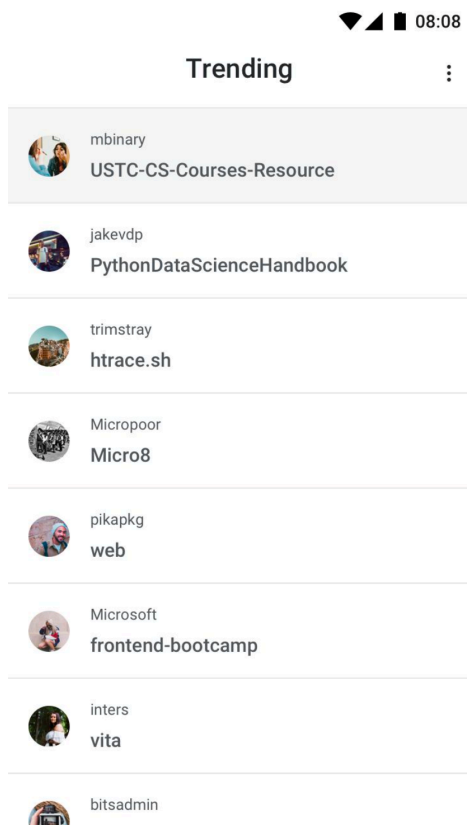
If you have any doubts anywhere on the assignment, feel free to get in touch with our recruiter. All the best!
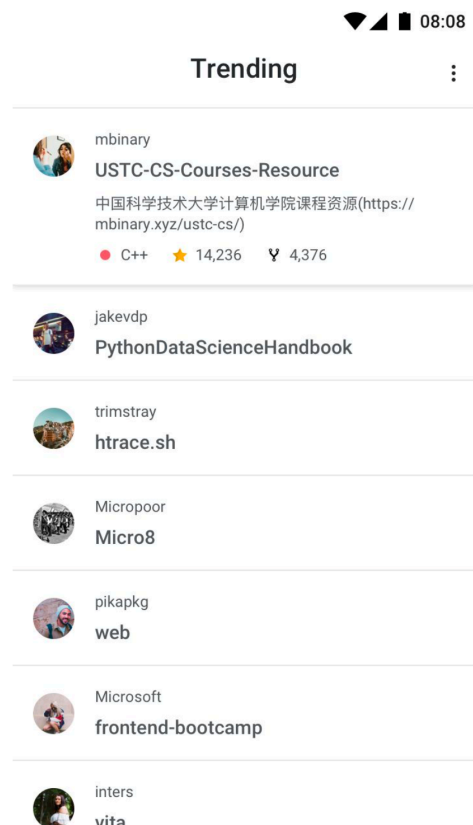
Trending ▼◢ ▮ 08:08

(Fig 1. Loading state while repositories are being fetched)

Trending ▼◢ ▮ 08:08

mbinary
USTC-CS-Courses-Resource

jakevdp
PythonDataScienceHandbook

trimstray
htrace.sh

Micropoor
Micro8

pikapkg
web

Microsoft
frontend-bootcamp

inters
vita

bitsadmin

(Fig 2. List of trending repositories)

Trending ▼◢ ▮ 08:08

mbinary
USTC-CS-Courses-Resource

jakevdp
PythonDataScienceHandbook

trimstray
htrace.sh

Micropoor
Micro8

pikapkg
web

Microsoft
frontend-bootcamp

inters
vita

bitsadmin

(Fig 3. Touch feedback on items)

Trending ▼◢ ▮ 08:08

mbinary
USTC-CS-Courses-Resource
中国科学技术大学计算机学院课程资源(https://mbinary.xyz/ustc-cs/)
● C++   ★ 14,236   ⑂ 4,376

jakevdp
PythonDataScienceHandbook

trimstray
htrace.sh

Micropoor
Micro8

pikapkg
web

Microsoft
frontend-bootcamp

inters
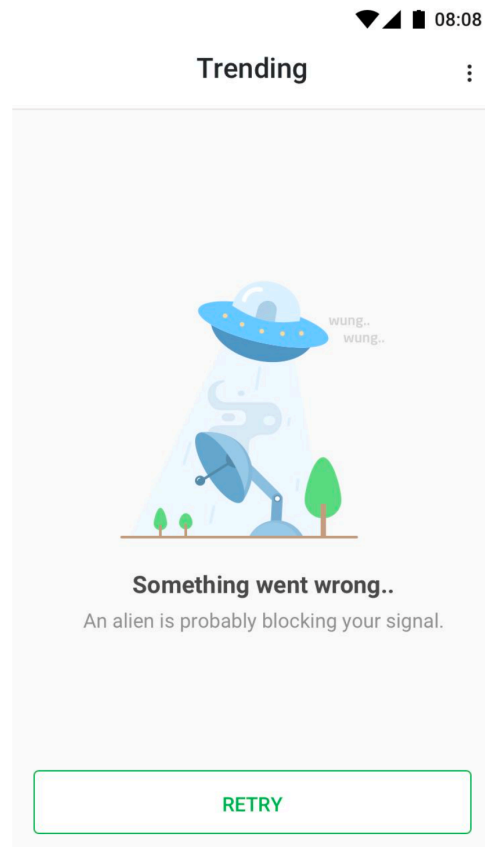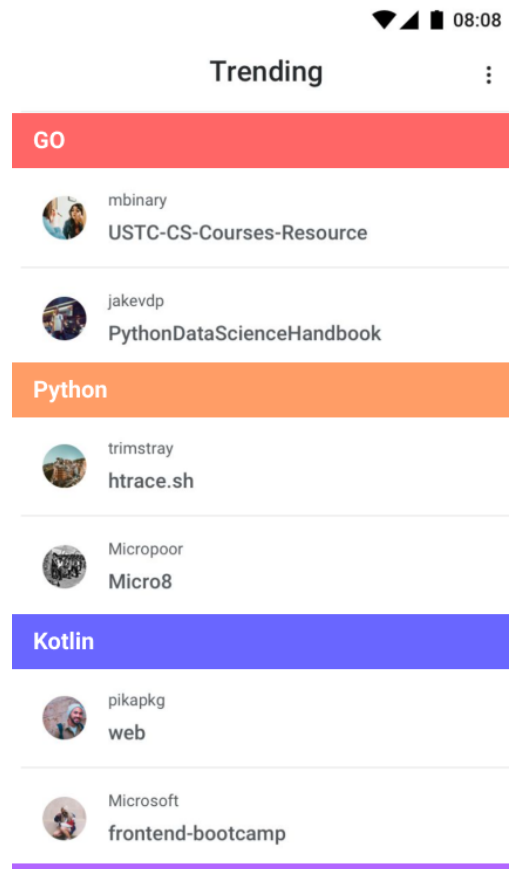vita

(Fig 4. Items can expand on being tapped)
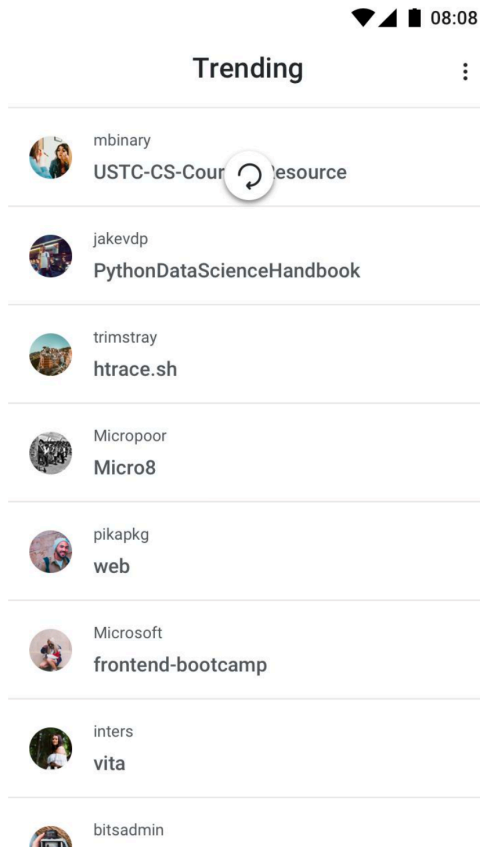
(Fig 5. Pull-to-refresh to fetch fresh data)

(Fig 6. Error state with retry option)

(Fig. 7. Headers based on repository language)