GitHub Username: Olga Kuklina

GitHubJourney

Description

A non-official GitHub mobile app helps users work with their github content, monitor github activity, check for news and updates.
The application implements secure login with OAuth to access user data.
In the main screen user is able to monitor contributions activity placed in chronological order.
The application allows user to monitor profile data such as feed list, repositories, followers/followings list, and starring activity.

Intended User

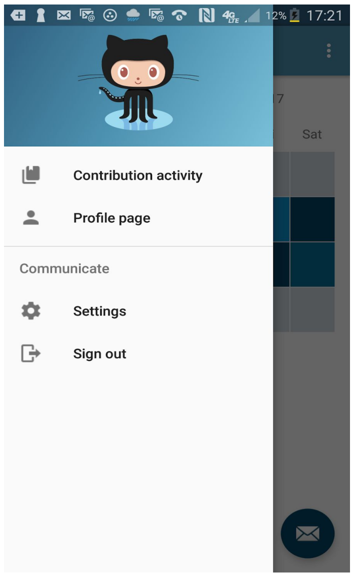
This app is developed for all Github users, who have github profiles.

Features

- The app provides access to user repositories with numbers of stars and forks.
- Allows user to:
  - See number stars and clones of your repositories
  - See list of followers of current user
  - See list of users followed by current user
  - Monitor feed list with different types of user activity.
  - See number of starred repositories with number of stars, clones and watchers.

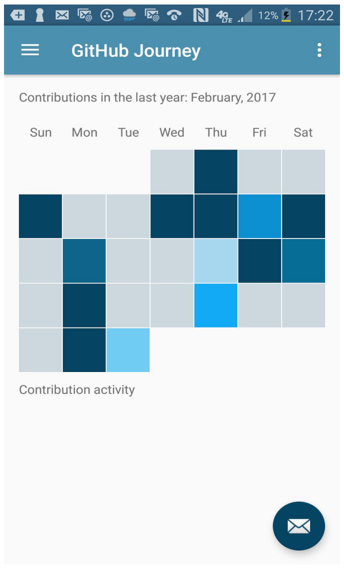
User Interface Mocks

Screen 1



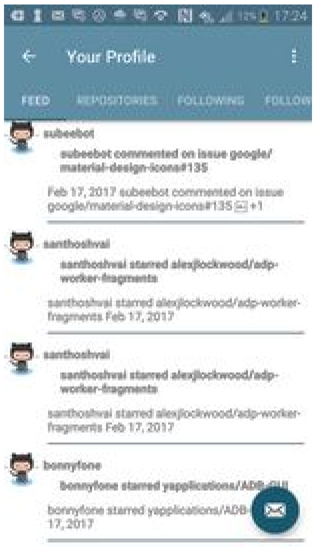
The navigation drawer provides a list of screens and actions to start journey.

Screen 2



The main screen UI is implemented as a calendar monthly view concept, which represents daily contributions activity. Each squared area implements a weekday with corresponding color displaying contributions intensivity.
By pressing on a weekday, user can open the contributions activity details for this particular day.

Screen 3



12% 17:24

Your Profile

FEED REPOSITORIES FOLLOWING FOLLOW

subeebot
subeebot commented on issue google/material-design-icons#135
Feb 17, 2017 subeebot commented on issue google/material-design-icons#135 +1

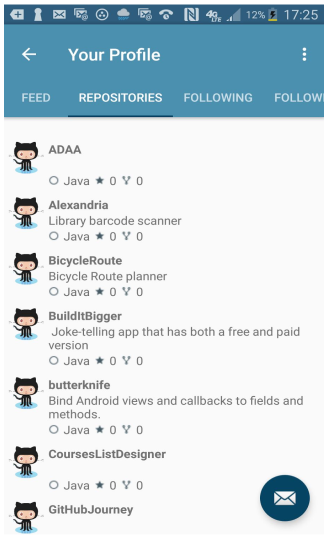
santhoshvai
santhoshvai starred alexjlockwood/adp-worker-fragments
santhoshvai starred alexjlockwood/adp-worker-fragments Feb 17, 2017

santhoshvai
santhoshvai starred alexjlockwood/adp-worker-fragments
santhoshvai starred alexjlockwood/adp-worker-fragments Feb 17, 2017

bonnyfone
bonnyfone starred yapplications/ADB-GUI
bonnyfone starred yapplications/ADB-
17, 2017

By clicking on user profile page on the navigation drawer screen, user will be transferred to a profile data representation screen.
The first tab represents feed list with author, date, activity title and description.

Screen 4



12% 17:25

Your Profile

FEED REPOSITORIES FOLLOWING FOLLOWI

ADAA
Java ★ 0 0

Alexandria
Library barcode scanner
Java ★ 0 0

BicycleRoute
Bicycle Route planner
Java ★ 0 0

BuildItBigger
Joke-telling app that has both a free and paid version
Java ★ 0 0

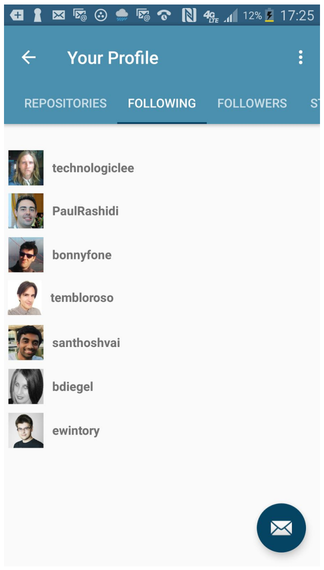
butterknife
Bind Android views and callbacks to fields and methods.
Java ★ 0 0

CoursesListDesigner
Java ★ 0 0

GitHubJourney

The second tab implements user repositories list with appropriate information such as language, number of stars and forks.

Screen 5



17:25

Your Profile

REPOSITORIES FOLLOWING FOLLOWERS

technologiclee

PaulRashidi

bonnyfone

tembloroso

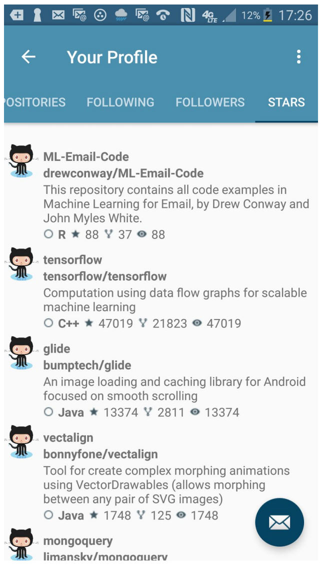
santhoshvai

bdiegel

ewintory

The next two tabs show list of users followed by the profile owner and list of user's followers with information about github nicknames and profile pictures.

Screen 6



17:26

Your Profile

OSITORIES FOLLOWING FOLLOWERS STARS

ML-Email-Code
drewconway/ML-Email-Code
This repository contains all code examples in Machine Learning for Email, by Drew Conway and John Myles White.
R ★ 88 37 88

tensorflow
tensorflow/tensorflow
Computation using data flow graphs for scalable machine learning
C++ ★ 47019 21823 47019

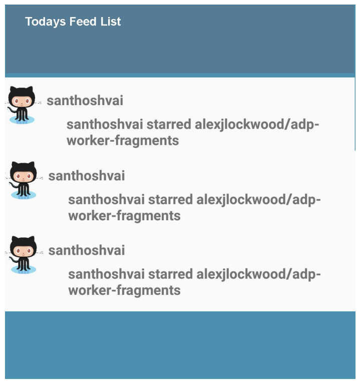
glide
bumptech/glide
An image loading and caching library for Android focused on smooth scrolling
Java ★ 13374 2811 13374

vectalign
bonnyfone/vectalign
Tool for create complex morphing animations using VectorDrawables (allows morphing between any pair of SVG images)
Java ★ 1748 125 1748

mongoquery
limansky/mongoquery

The last tab represents user starred repositories with repository titles and descriptions, language information, number of forks and watchers.

Screen 7


Todays Feed List

santhoshvai

santhoshvai starred alexjlockwood/adp-worker-fragments

santhoshvai

santhoshvai starred alexjlockwood/adp-worker-fragments

santhoshvai

santhoshvai starred alexjlockwood/adp-worker-fragments

The app widget provides a list of today's feeds and invitations to start journey: by clicking on a list item, the app opens on the feed list tab.

Key Considerations

How will your app handle data persistence?

The Github api serves user activity data only for three recent months. In order to avoid requesting all three months every time user opens the app, it will store already loaded user contributions into a local SQLite database. The app will maintain an embedded content provider to access the contribution data and request GitHub api to send only most recent activity records.

Describe any corner cases in the UX.

The Application maintains support of backward navigation between different screens by pressing the Back button.

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

The app will use:

- Picasso for loading and rendering images
- CommonsIO/CommonsHttpClient for accessing GitHub REST api
- A number of standard Android libraries

Describe how you will implement Google Play Services.

This app integrates Google Play Maps and Firebase Analytics.

- The app creates a map view to display followers/following users locations (for those, who provided this information).
- The app emits analytics events for main UI scenarios.

Next Steps: Required Tasks

Task 1: Project Setup

- The app uses GitHub API. Get your API keys in order to run the app.
- Create a new resource file using the following path: app/src/main/res/values/client_credentials.xml.
- Put "client_id" and "client_secret" key values in it.
- Application min API Level: 22, Android 5.1 (LOLLIPOP_MR1)

Task 2: Implement UI for Each Activity and Fragment

List the subtasks:

- Build UI for user login activity screen
    - Implement authentication async task
    - Implement login screen layout
    - Implement authentication activity
- Implement main activity with a navigation drawer
- Build UI for Contributions activity screen
    - Implement fragment_main layout with grid layout
    - Implement grid list item layout
    - Implement grid list empty item
    - Implement MainView fragment
    - Implement user activity async task based loader
    - Implement contributions list adapter
    - Create a user activity data entry class

- Implement user profile page UI
    - Implement general screen layout with tabs
        - Implement General activity with View pager
        - Reuse the above user activity data entry class
        - Implement feed list UI
            - Implement feed list layout
            - Implement feed list async task based loader
            - Implement feed adapter
            - Create a feed data entry class
        - Implement repositories UI
            - Implement repositories list layout
            - Implement repositories async task based loader
            - Implement repositories adapter
            - Create a repositories data entry class

        - Implement followings UI
            - Implement followings list layout
            - Implement followings async task based loader
            - Implement followings adapter
            - Create a github user data entry class
        - Implement followers UI
            - Implement followers list layout
            - Implement followers async task based loader
            - Implement followers adapter
            - Reuse the github user data entry class
        - Implement stars UI
            - Implement star list layout
            - Implement stars async task based loader
            - Implement stars adapter
            - Create a stars data entry class

Task 3: Settings and Logout

- Build UI for settings screen
    - Implement settings screen activity
- Implement logout by selecting a respective item in the navigation drawer.
    - Implement an async task for logout

Task 4: User contribution backend

User contribution storage concept:

- Design a SQLite database to hold user contributions for 1 year
- Implement a db helper class to talk to the database
- Implement a content provider to access the contributions

Task 5: Extending the contribution calendar

Making the contribution calendar responsive to user actions

- Create layout to show contribution list for given day
- Implement a click listener for all days that have contributions
- Implement talking to content provider to query contribution data

Task 6: Implement a “take screenshot” feature

Make a screenshot of a screen and share it by email

- Implement a click listener for floating action button
- Implement taking a screenshot
- Implement sending by email

Task 7: Implement an “app widget”

Implement a widget to provide relevant information to user on the home screen.

- Implement app widget provider
- Implement app widget layout and metadata
- Implement app widget activity

Task 8: Extending the following/followers tab with a map view

- Plug Google Play Map Service to the project
- Implement a map screen to display following user locations
- Implement a map screen to display follower user locations

Task 9: Adding analytics

Add some essential analytics functionality to track how users interact with the app.

- Plug Firebase Analytics to the project
- Implement metrics to track main user actions
    - Switching between activities
    - Switching between tabs on the same activity

Task 10: A future task for version 2.0

Make user nicknames and repository names clickable

- Click to a user nickname will open a profile screen for this user
- Click to a repository name will open a screen with the repository details
- Investigate how much I can reuse the existing functionality and ui layouts to display details of other github users (not only profile owner)
- Displaying repository details will require new layouts, activities/fragments and an async task etc