
GitHub Username: xiaoyifan

MeditReader

Description

Not sure how to write a good description? Search 5-star apps on the Play Store for inspiration. This is an app which provides users with a connection of sharing minds and inspiration. It's also a platform to provide what you are thinking, and what you learnt, experienced. Write a story, and we all can see it.

Intended User

Everyone who is passionate about design, technology, lifestyle, society and needs some inspiration.

Features

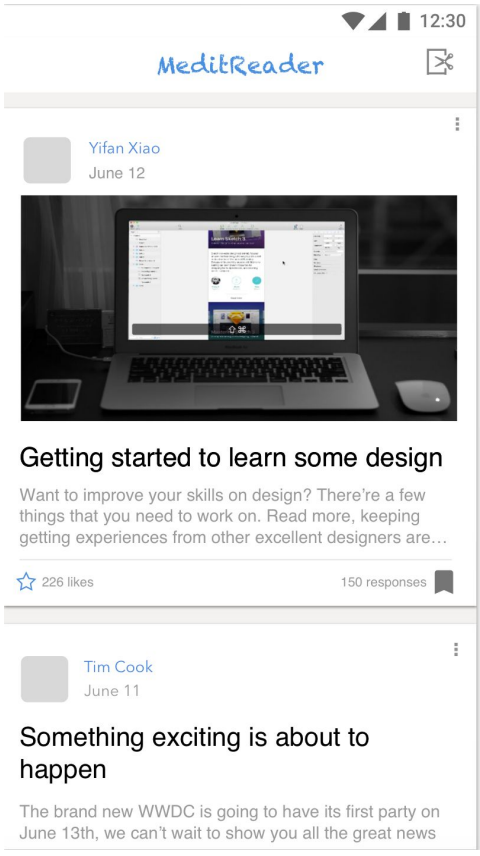
Main features of the app:

- Publish articles, with images and tags attached.
- Pick an image from album or take one with your camera
- Get feed articles that you might care about
- React to the articles with comments and likes

User Interface Mocks

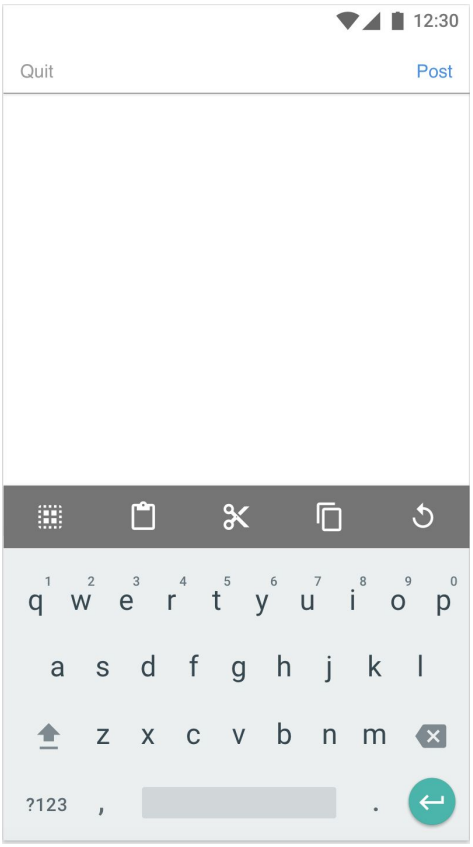
These Mocks are created by using Sketch.

Screen 1 Main Screen



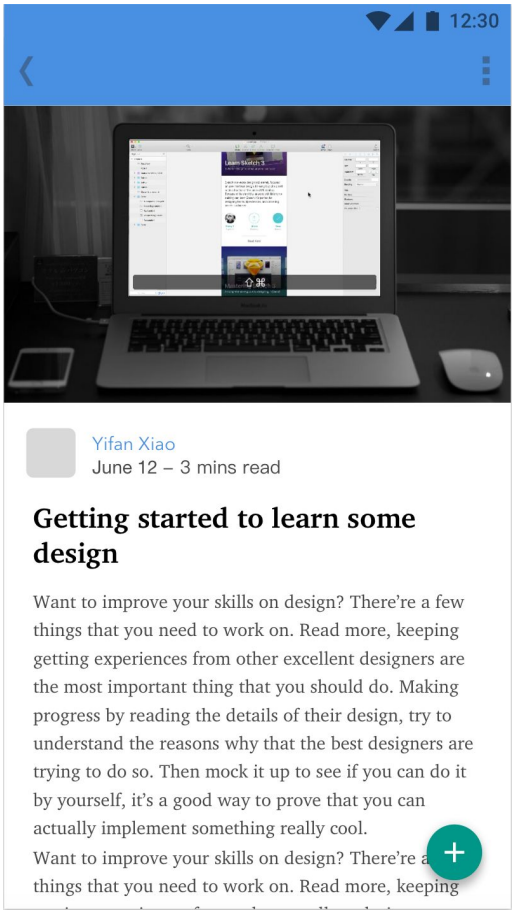
The Main Screen are basically presents users with a list of articles. Users could scroll up and down to view more.

Screen 2 Edit Screen



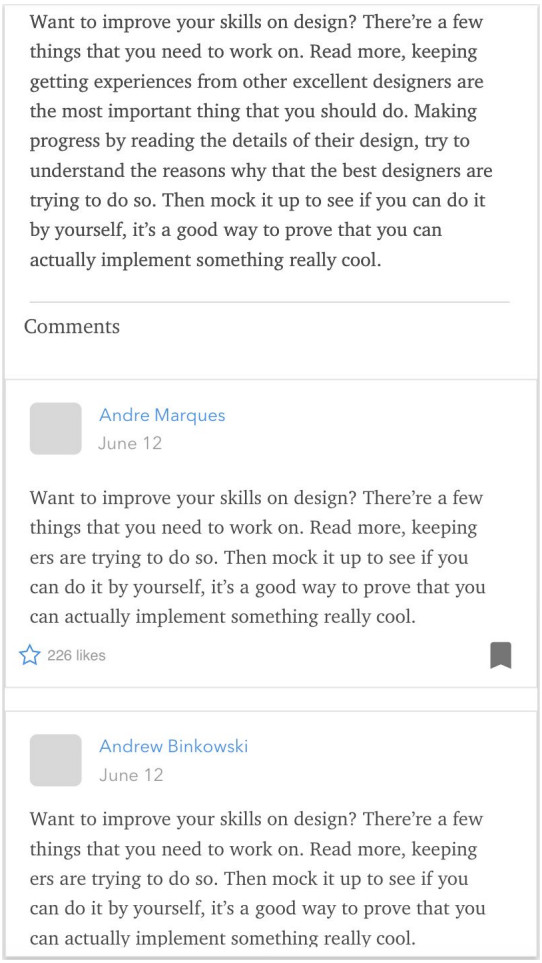
Edit Screen Provides users with a edit interface for blog editing.

Screen 3 Detail Screen-1



Detail Screen-1 is what you will see when you get in the detail view. With Nav Bar on the top, and when you scroll up to view more details about the article, The image will disappear in a parallel way, and a FAB will show in the bottom-right side of the screen, tap on it you can comment it.

Screen 4 Detail Screen-2



Keep scrolling down up the detail view to the end, you will be able to the comments from other users.

Key Considerations

How will your app handle data persistence?

The app will use firebase as online data storage, also the app needs an content provider and sqlite database as the strategy to cache data locally when there's no network available.

Describe any corner cases in the UX.

The article grid/ recycler view will be used to display the intro of articles, and when the user tapped on the article item, we will use the image shared transition as the animation to display the article. And a floating button will display on the screen when the image is completely hidden for users to be able to make a comment

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

ButterKnife, for quick UI elements binding in the code.
Okhttp, for networking call
Design, for UI animations
Picasso, for image loading async loading
// to be added more, if there's anything necessary to be used in the project

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

Init the Android project and setup the code base. Configure the project with necessary libraries. Make the basic setup is right and can work:

- Initiate the project with Android Studio
- Add libraries necessary to this project, and snyc with Gradle.

Task 2: Implement UI for Each Activity and Fragment

All the substasks are listed below:

- Make precise Mockups for the every screens
- Import all the UI element assets to the project
- Build the layout of main screen (main activity)
- Build details for detail activity
- Add floating buttons, sharing buttons and other buttons on the navgation bar.

Task 3: Backend Configuration

This task is basically the configuration of Google Firebase, include all the required API keys to the project, and make sure Firebase works well.

Subtasks:

- Create the app in Google Firebase
- Integrate the API Keys and configuration files into the app
- Testing to make sure it works well, like login and data fetching

Task 4: Model implementation and local backend implementation

Design model objects and content provider to support local data fetching:

- Design data model for articles, comments
- Implement content providers and DBHelper
- Implementation and robustness testing

Task 5: Connection

Connect the backend with UI, to make the app fully working:

- Add methods to present all the data we get from backend to the UI
- Make necessary improvement
- Add extra features like floating buttons for sharing and making comments

---