How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: "**Capstone_Stage1**"
3. Replace the text in green

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"

GitHub Username: vclub

TesterHome App

Description

A simple, open source, TesterHome.com Chinese community reader application. Read topics of TesterHome.com on the mobile devices you carry around daily! This application is built for both phones and large tablets, and is designed using Google's Material Design guidelines.

Intended User

This app is primarily aimed at people who wish to track Tester Home BBS.

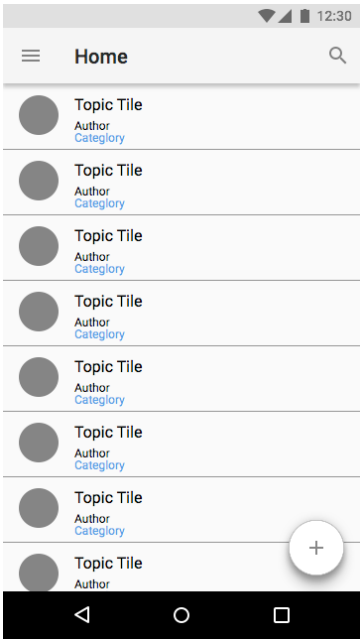
Features

- Show last topics of TesterHome
- Post new topic to TesterHome
- Open web link with customtabs
- Uses ShareActionProvider to share content

User Interface Mocks

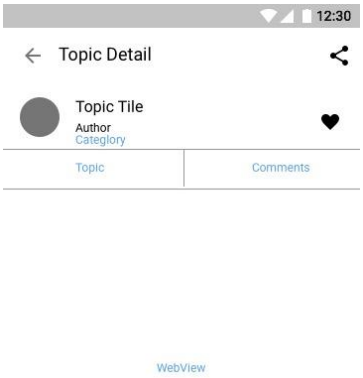
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



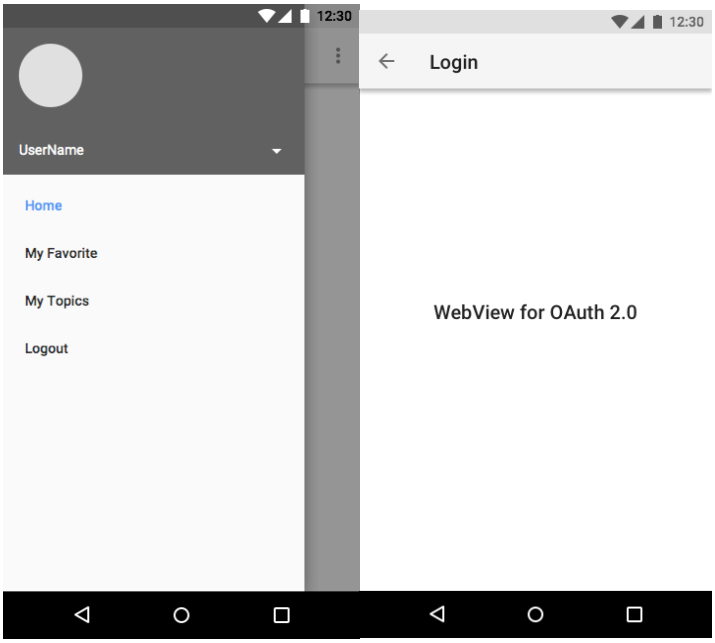
This is the main page of the app, show a list of last topics.

Screen 2



This is the detail of the topic, if user login, they can add to favorite list, user can share content by click the share button on the right top.

Screen 3



User can use drawer layout to login , logout,

Key Considerations

How will your app handle data persistence?

All information are get from TesterHome.com API. The favorite info store in local SQLite database with a ContentProvider.

Describe any corner cases in the UX.

The topics (screen 1) and topic detail (screen 2) screens will appear in a Master-Detail design pattern between phones and tablets.

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

The project depends of libraries like ButterKnife UI dependency injection, Gson to handle Json data get from API, Retrofit to API requests, Fresco to handle the loading and caching of images.

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

Setup project on Android Studio, configure library and start to manage the tasks.

- Create and configure project on Android Studio
- Configure libraries
- Create BaseActivity extends AppCompatActivity and with toolbar

Task 2: Implement UI for Each Activity and Fragment

Implement the main screen with all related screens.

- Build UI for MainActivity
- Build UI for TopicsFragment
- Build UI for TopicDetailFragment
- Uses ShareActionProvider to share content with an outside application

Task 3: Implement OAuth2.0 login

Implement login UI

- Build UI for LoginActivity
- Build UI for user profile

Task 4: integrates Admob and Analytics Google Services.

- Create paid flavore for app, remove admob from paid app
- Add Analytics Google Service

Task 5: Generate APK, Prepare for deployment

- Generate app flavor keys
- Create APK

Add as many tasks as you need to complete your app.

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"