GitHub Username: NoSimple

MyPetNews

Description

Follow the news from different communities. Get up-to-date information every day. Read news, events, and comments on them.

Intended User

This app is primarily intended for people who want to follow news and events in the IT world.

Features

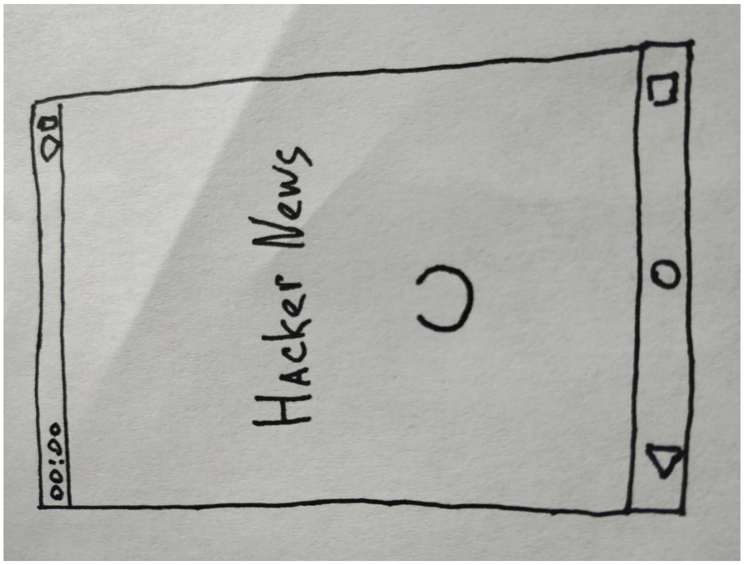
- The app will use the Java language for development purposes.
- The app will utilize all stable release versions of libraries, Gradle, and Android Studio.
- Android Studio v. 3.6.2
- Gradle v. 6.3
- AndroidX v. 1.1.0
- Material v. 1.1.0
- The app will use Butterknife - 10.2.1
- The app will use Retrofit2 - 2.8.1
- The app will use Picasso - 2.71828
- The app will use LifeCycle - 2.2.0
- The app will use Espresso - 3.2.0

List the main features of your app. For example:

- Shows the latest news.
- Shows comments on news.
- Allows you to view videos.
- Allows you to select news by topic.
- Allows you to share the news with other people.
- Have a widget display bookmarked events.

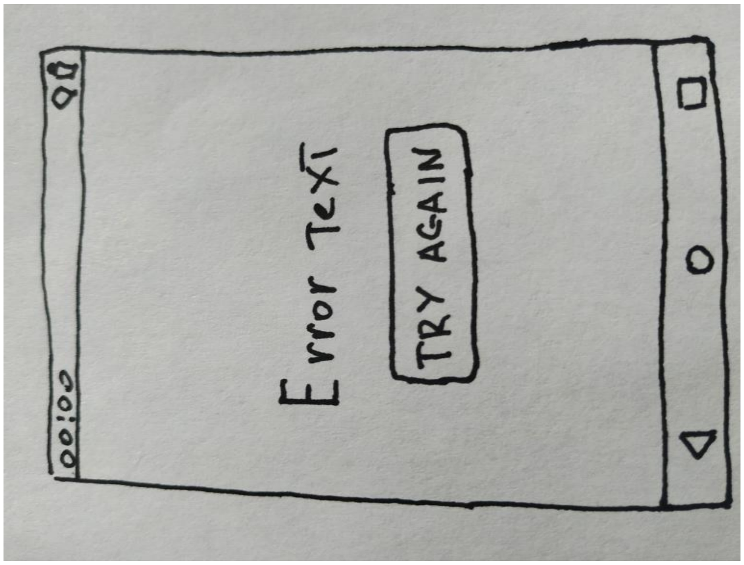
User Interface Mocks

Screen 1



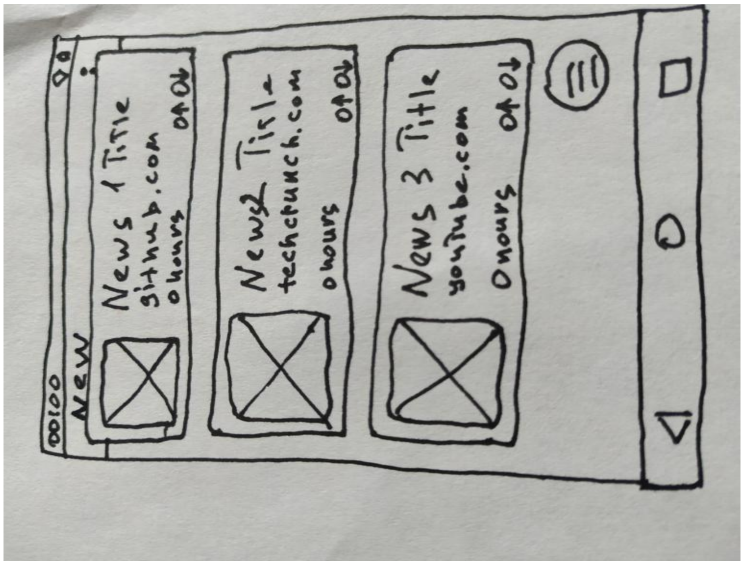
Primary data is loaded.

Screen 2



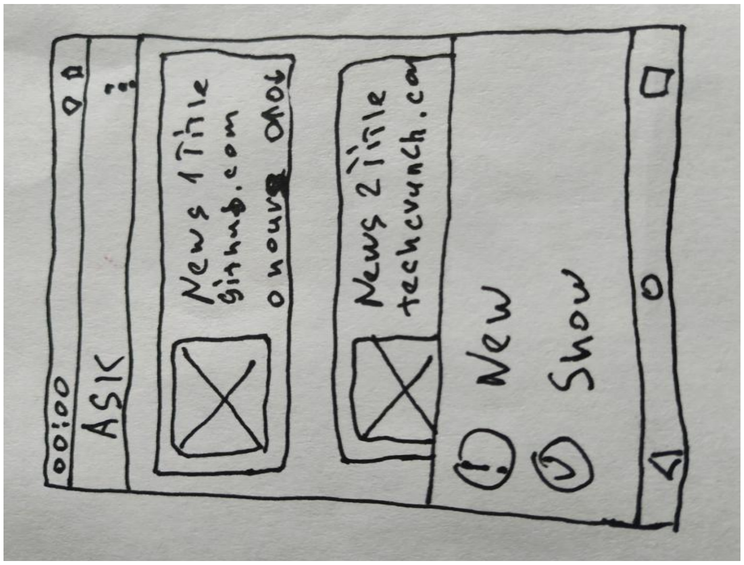
If there is an error, we show it on the screen and allow the user to reload the data.

Screen 3



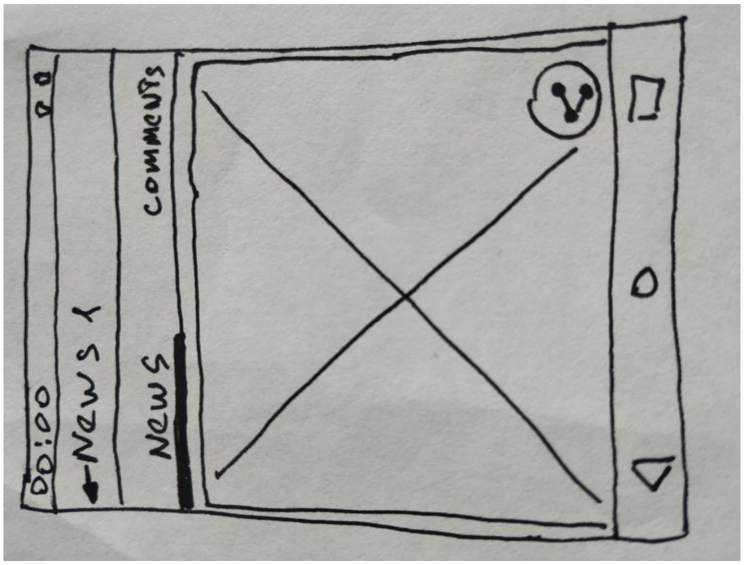
If the data is received, we show it to the user.

Screen 4



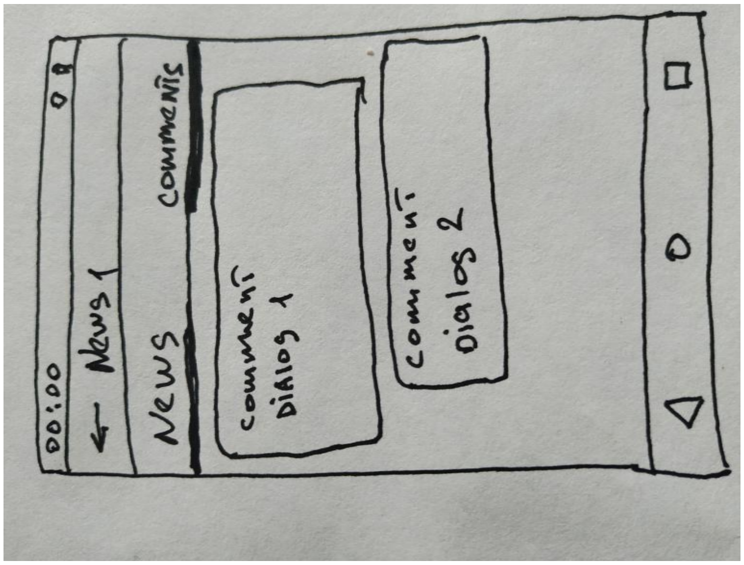
Allow the user to select a theme using the bottom dialog.

Screen 5



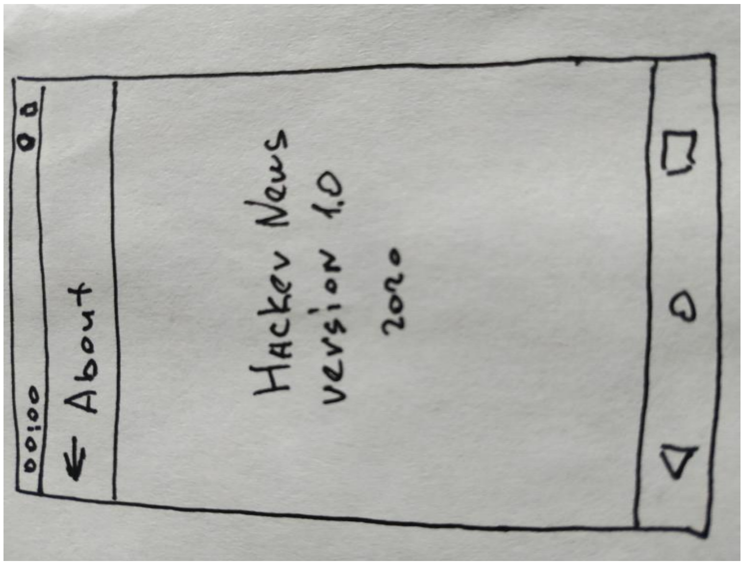
Showing the user the selected news item.

Screen 6



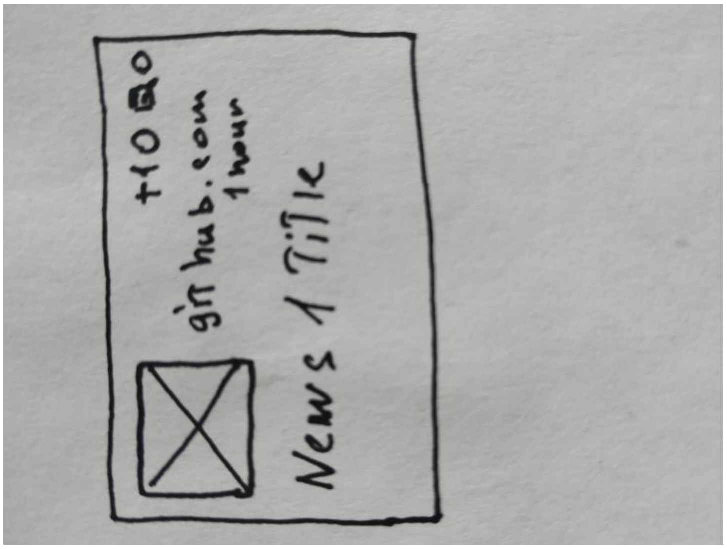
Showing the user comments on the news, if there are any.

Screen 7



Screen with information about the app.

Screen 8



Widget screen.

Key Considerations

How will your app handle data persistence?

- Live Data to keep the current search result open.
- SavedInstanceState to temporarily save the user input.
- SharedPreferences to save the search criteria.

- App use AsyncTask to download a specific news item.

Describe any edge or corner cases in the UX.

Pages, with the exception of Splash, provide "reverse" navigation to return to the parent action. There are also floating action button for switching to the bottom dialog.

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

Picasso for image loading. Butterknife for findViewById calls. Retrofit for network calls. AndroidX Preference Library for the settings.

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

Google Analytics for detailed statistics of app users.

Next Steps: Required Tasks

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

Task 1: Project Setup

Write out the steps you will take to setup and/or configure this project. See previous implementation guides for an example.

- Configure libraries.
- Identify dependencies.
- Planning and implementing a skeleton for architecture.
- Connection to Google Analytics.

Task 2: Implement UI for Each Activity and Fragment

- Implement Data provider.
- Put strings, colors and styles in the strings.xml, colors.xml and style.xml
- Build data models.
- Build UI

Task 3: Implement SplashActivity

- Build UI for SplashActivity.
- Create ViewModel.

Task 4: Implement MainActivity

- Build UI for MainActivity.
- Implement RecyclerView item.
- Create ViewModel.
- Create BottomDialog.

Task 5: Implement AboutActivity

- Build UI for AboutActivity.

Task 6: Implement StoryActivity and Fragments

- Build UI for StoryActivity.
- Create ViewModel.
- Create fragments.
- Create WebView.

Task 7: Implement App Widget

- Implement App Widget.
- Build UI for main widget.