
GitHub Username: jgroman

Hacker News Client

Description

This app provides seamless and comfortable access to the latest IT industry news and relevant discussions on a web site Hacker News - https://news.ycombinator.com/

Intended User

The app is best suited for computing industry professionals and also enthusiasts who like to stay informed about the current developments in IT and related industries.

Features

App displays Hacker News items from various provided feeds provided by the server. Feeds include:

- New Stories

- Top Stories
  Best Stories
- Ask Stories
- Show Stories
- Job Stories

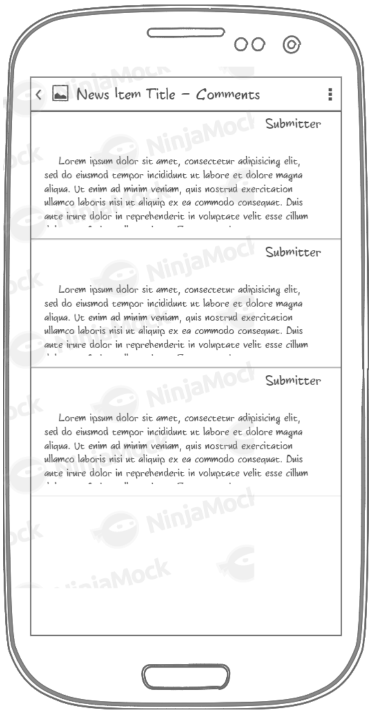
App provides means to display original linked article and relevant discussion. App also enables user to upvote selected news items and display selected user's profile.

User Interface Mocks

Stories List

Application entry point, displays list of stories from selected news feed. It is possible to select news feed by using top tab bar or swiping sideways.

Story Comments



Clicking on story list item opens list of story comments. Every comment contains submitter name and comment text.

Story link - article

Long click on story list item opens original linked article in system WebView or optionally in custom Chrome tab.

App Widget

Widget displays New Stories news items only. On-click behavior of widget list items is the same as on Stories List screen. Widget content is periodically refreshed with new data downloaded from XML feed.

Key Considerations

How will your app handle data persistence?

The app will use local Content Provider based on SQLite database for caching feed and user data and SharedPreferences to store local user settings.

Describe any edge or corner cases in the UX.

App requires connection to the internet, users will be notified if no connection is available. Tablet UI will enable seeing both news items and their content at the same time. Care must be taken to correctly handle display orientation changes or unexpected activity/fragment state changes.

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

Android Support Library V27.1.1 - Material Design support
OkHttp library V3.10.0 - to simplify fetching of XML feeds.

Stetho debug bridge V1.5.0 - for app debugging

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

App will use Google Sign-In to simplify the sign-in experience and Google Analytics to get insight on app usage.

Design Specification

- App is written solely in the Java Programming Language.
- App keeps all strings in a strings.xml file and enables RTL layout switching on all layouts.
- App includes support for accessibility - including content descriptions, navigation using a D-pad, and , if applicable, non-audio versions of audio cues.
- App provides a widget to provide relevant information to the user on the home screen
- App implements at least one of the three:
  - If it regularly pulls or sends data to/from a web service or API, app updates data in its cache at regular intervals using a SyncAdapter or JobDispacter.
  OR
  - If it needs to pull or send data to/from a web service or API only once, or on a per request basis (such as a search application), app uses an IntentService to do so.
  OR
  - It it performs short duration, on-demand requests(such as search), app uses an AsyncTask.

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

- Create new empty Android App in Android Studio
- Add all project dependencies into app-level build.gradle configuration
- Perform Gradle sync

Task 2: Implement UI for Each Activity and Fragment

- Build UI for StoriesListActivity
- Build Fragments for each story type
- Build StoryWebViewActivity
- Build StoryCommentsActivity
- Create SharedPreferences screen
- Create App Widget to show New Stories

Task 3: Create Hacker News XML feed parser

Write XML feed parser implementation which would be capable of correctly parsing all provided types of Hacker News feed: New, Top and Best Stories, Ask HN, Show HN, Jobs

Task 4: Implement feed item fetcher

Feeds may contain diverse types of items, implement fetching mechanism which would correctly recognize and obtain all information related to the item.

Task 5: Display retrieved data in app UI

Correctly show downloaded information at the appropriate places in app UI.

Task 6: Implement alternative layouts

Create layouts and make necessary code changes for correct display in landscape mode and on tablets.

Task 7: Create App widget

Implement SQLite content provider and create widget for displaying New Stories feed.

Task 8: Create build variants

Create Debug and Release app variants. Remove all debug related parts from Release variant.