Capstone_Stage1

GitHub Username: aac9095

AAC News (link: github.com/aac9095/AAC-News)

Description

"We become the books we read." - Matthew Kelly

A single place to easily read all the news you rely on to think, learn, and keep ahead. AAC News is a simple and easy to use app that gives you a dosage of everyday news. It displays headlines currently published on a range of news sources and blogs (26 and counting so far). Get the whole picture the whole time with this app. Specially crafted for tablets and phones, this app brings you breaking news and full access content of several news sources.

Intended User

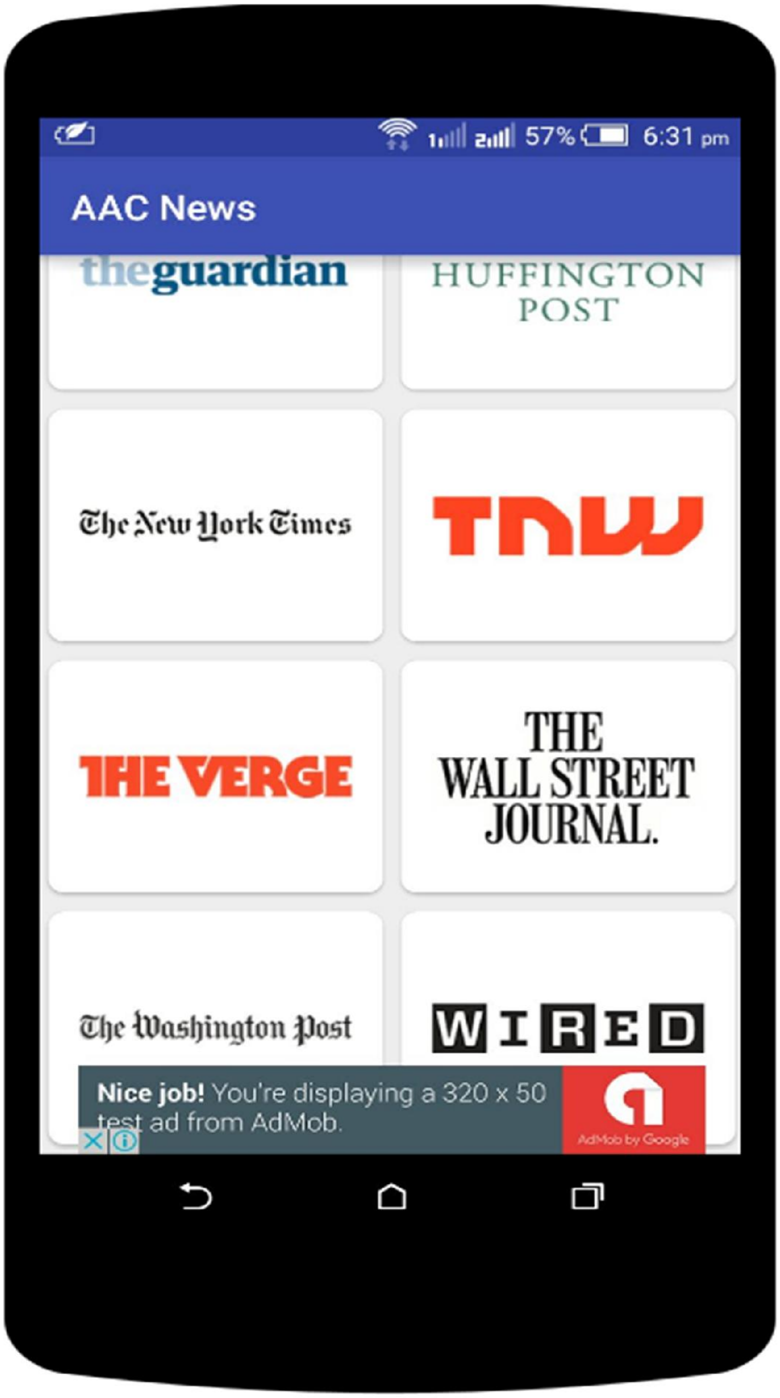
Anyone who wants a daily dose of good news

Features

- Select from various news sources and blogs
- Get summarise news in 50-60 words
- Freaking Fast

User Interface Mocks

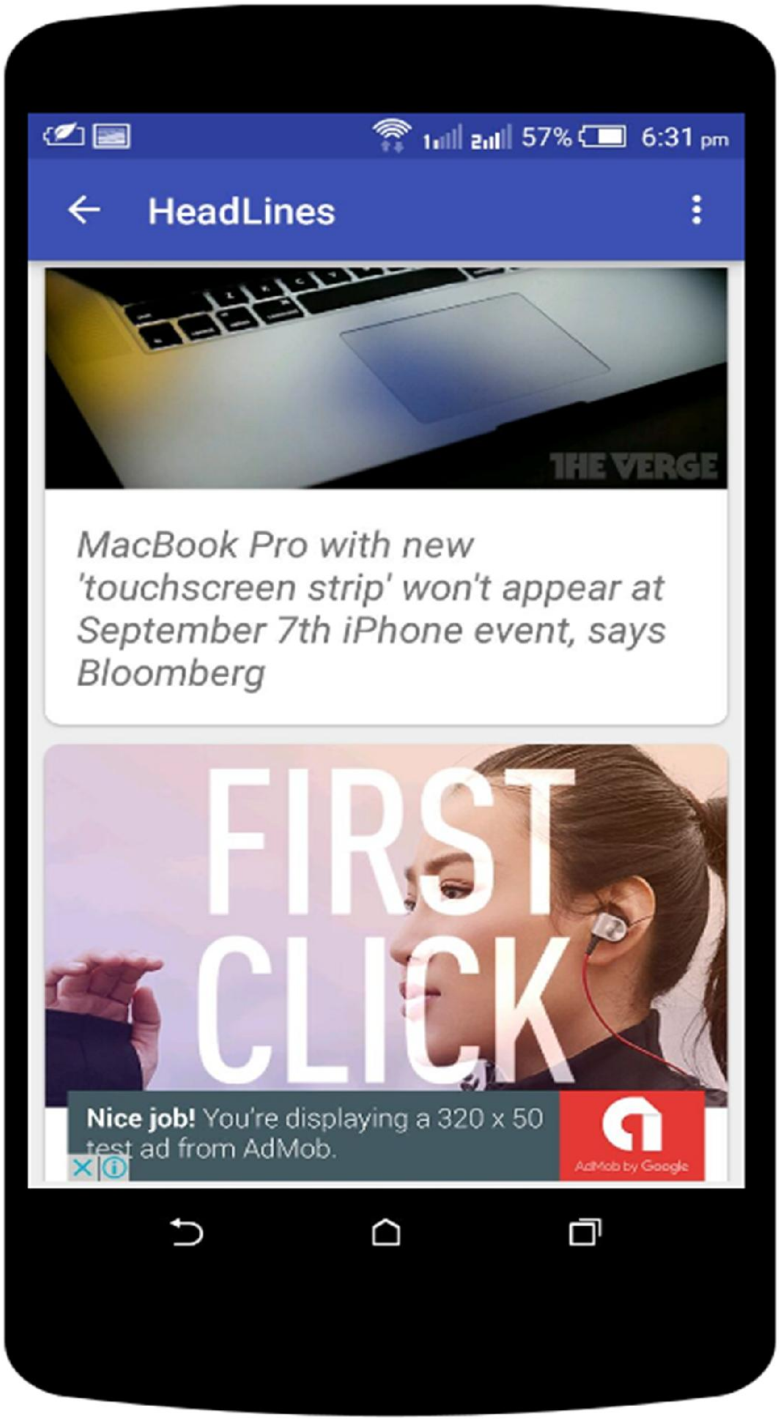
Screen 1



Nice job! You're displaying a 320 x 50 test ad from AdMob.

Launcher Activity of the app. In this activity user selects the new source or blog whose news he/she wants to read.

Screen 2

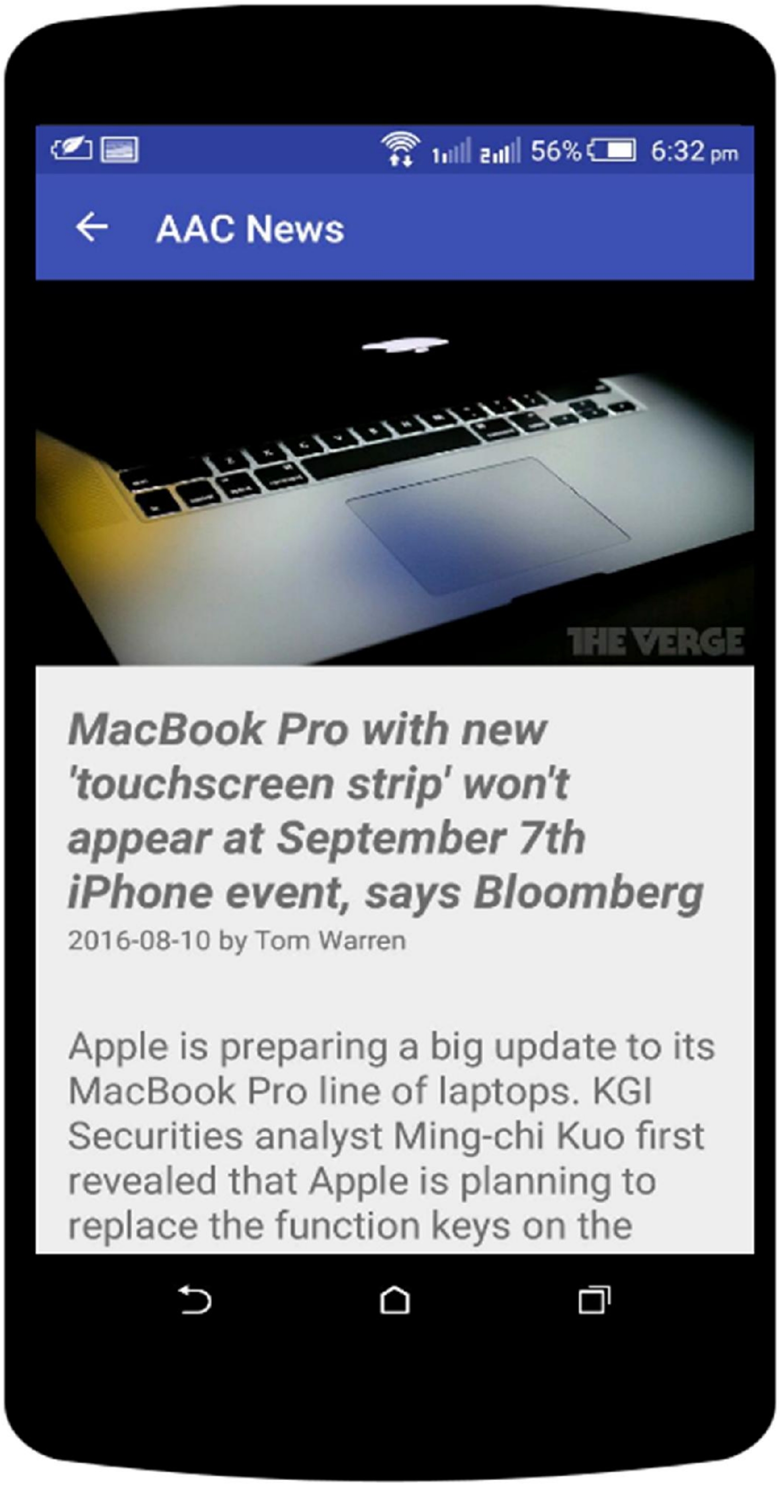


MacBook Pro with new 'touchscreen strip' won't appear at September 7th iPhone event, says Bloomberg

Nice job! You're displaying a 320 x 50 test ad from AdMob.

When the user select the new source/blog this activity is opened. It will contain the list of different news which can be sorted in mainly three ways i.e. Top, Recent or Popular

Screen 3



MacBook Pro with new 'touchscreen strip' won't appear at September 7th iPhone event, says Bloomberg

2016-08-10 by Tom Warren

Apple is preparing a big update to its MacBook Pro line of laptops. KGI Securities analyst Ming-chi Kuo first revealed that Apple is planning to replace the function keys on the

It is the detail news activity which is opened when user select a particular news. It contain summary of the news in 50-60 words and link to full news

Key Considerations

How will your app handle data persistence?

I will build the database that will store the news and news sources/blog and will use content provider to provide an abstraction between my data and UI layer and use Loaders to retrieve data efficiently. 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.

Describe any corner cases in the UX.

If the user rotate the screen during the async task than I will cancel any currently running task, save its state and start a new one with the saved state when the new Activity is created or I can also use onRetainNonConfigurationInstance() to pass your AsyncTask to the new Activity

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

Picasso – I will use this library to display the logo of news sources/blog or the image for the news.
Retrofit – I will use retrofit to turn my HTTP API into a Java interface.
Butter Knife – I will use Butter Knife to find and automatically cast the corresponding view in my layouts.
Google Play Services – I will use google play services to implement adMob and analytics.

Describe how you will implement Google Play Services.

Google Play Services: AdMob – I will use this service to display banner ads in every activity and interstitial ad whenever user select a news source/blog.

Google Play Services: Analytics – I will use this service to track user activity that which news source/blog the user viewed the most, error & crash reporting and lot more. Right now I have been thinking what more to do with this service, so will add more analytical features

Next Steps: Required Tasks

Task 1: Project Setup

- Update Android Studio to the latest stable version
- Update SDK
- Add dependencies for Picasso, Retrofit and Buttter Knife
- Get API key for News API from **newsapi.org**
- Update Manifest file for SyncAdapters and Google play services.
- Add permission for accessing internet and checking network state in Manifest

Task 2: Implement UI for Each Activity and Fragment

Subtasks for the project:

- Setup basic UI structure
- Build top part of each activity/fragment (AppBar, coordinatorLayout)
- Build Master/Detail for Tablet Support
- Build UI elements

Task 3: Implement Data Models and Content Provider

Build up the data models, events and groups, and implement data persistence.

Subtasks:

- Create data model classes
- SQLite database setup
- Implement Retrofit call in SyncAdapter and store the result it in the database
- Loader/adapters and UI

Task 4: Google Play Services

Subtasks:

- Track crashes and other stuff Analytics give us for free
- Track the news source/blog the user viewed the most
- Implements banner and interstitial ads

Task 5: Handle Error Cases

Subtasks:

- The phone does NOT have Internet connectivity
- The app starts a network request and then the activity is restarted because of an orientation change
- The task cancelled correctly

Task 6: Others

Subtasks:

- Widget
- Accessibility
- RTL

Task 7: Testing

Subtasks:

- Rotation
- Phone vs. Tablet
- Performance