- Description
- Intended User
- Features
- User Interface Mocks
  - Screen 1
  - Screen 2
- Key Considerations
    - How will your app handle data persistence?
    - Describe any corner cases in the UX.
    - Describe any libraries you'll be using and share your reasoning for including them.
    - Describe how you will implement Google Play Services.
- Next Steps: Required Tasks
  - Task 1: Project Setup
  - Task 2: Implement UI for Each Activity and Fragment
  - Task 3: Implement Data Layer
  - Task 4: API Integration
  - Task 5: Final steps


**GitHub Username**: karthikraj-duraisamy

# NewsAgent

## Description

NewsAgent is an Android application which aggregates all the latest news of categories like General, Entertainment, Sports, Science & Technology and etc..

Additionally NewsAgent application provides the ability to like and store the news content for future reference. User can share the news content.

## Intended User

Anyone who is interested to read news on the go in their Android smartphones.

# Features

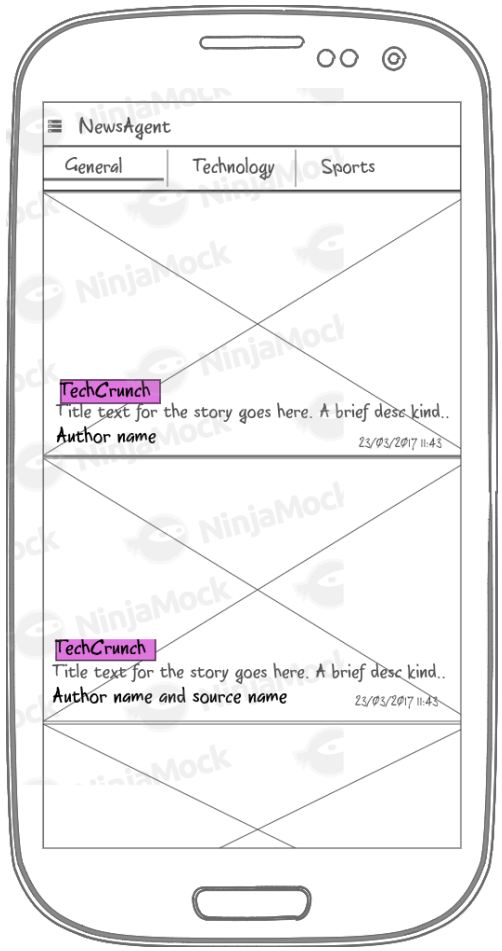
- Displays news feeds
- Category based selection of feeds
- Like and access offline feeds

# User Interface Mocks

### Screen 1

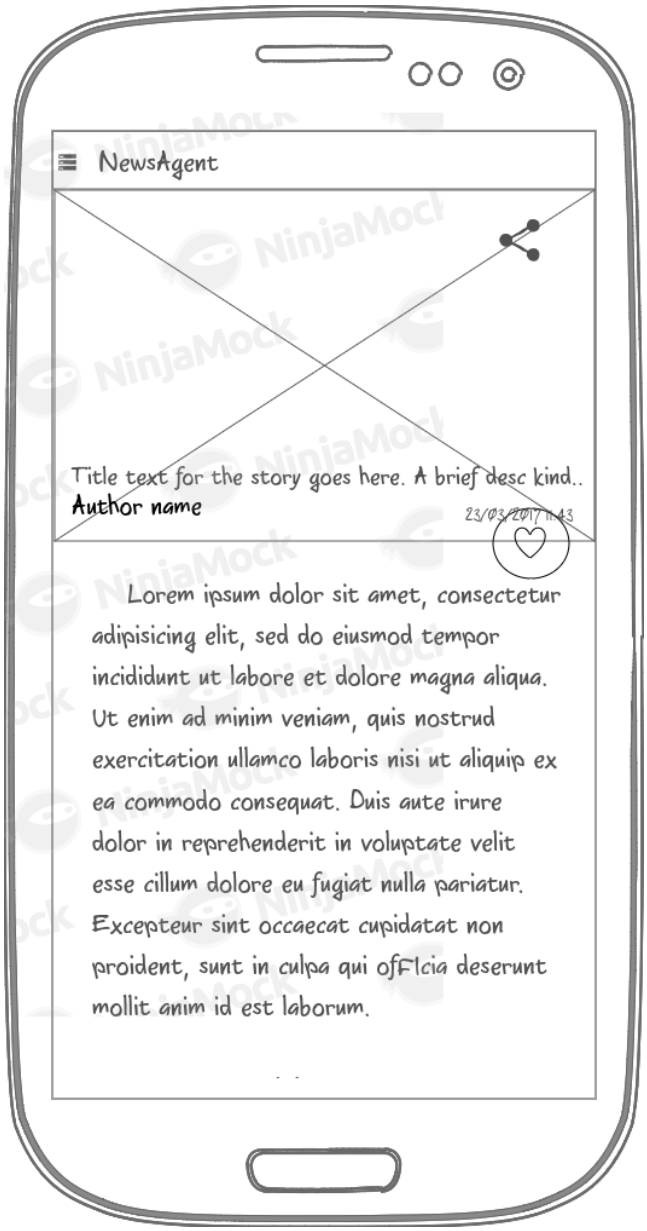

Navigation drawer menu and top level menu items screen.

## Screen 2



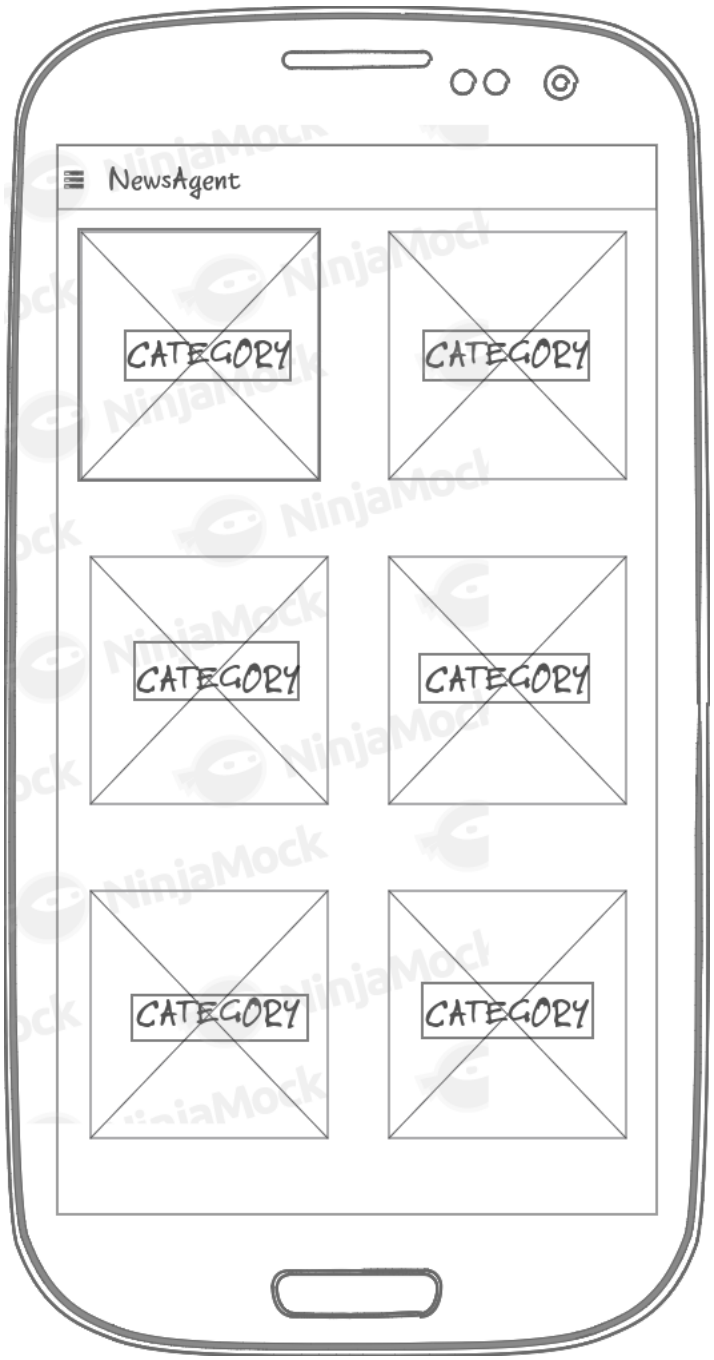

News Feed screen. This is going to be the home screen as well.

## Screen 3



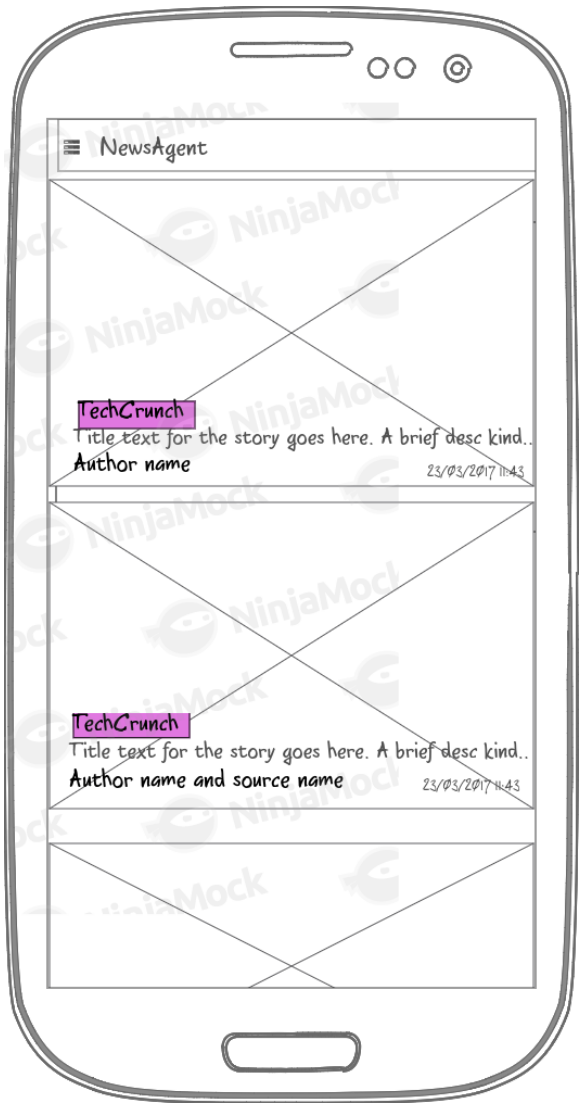

News feed details page screen. This screen will appear on click of the news feed item.

## Screen 4



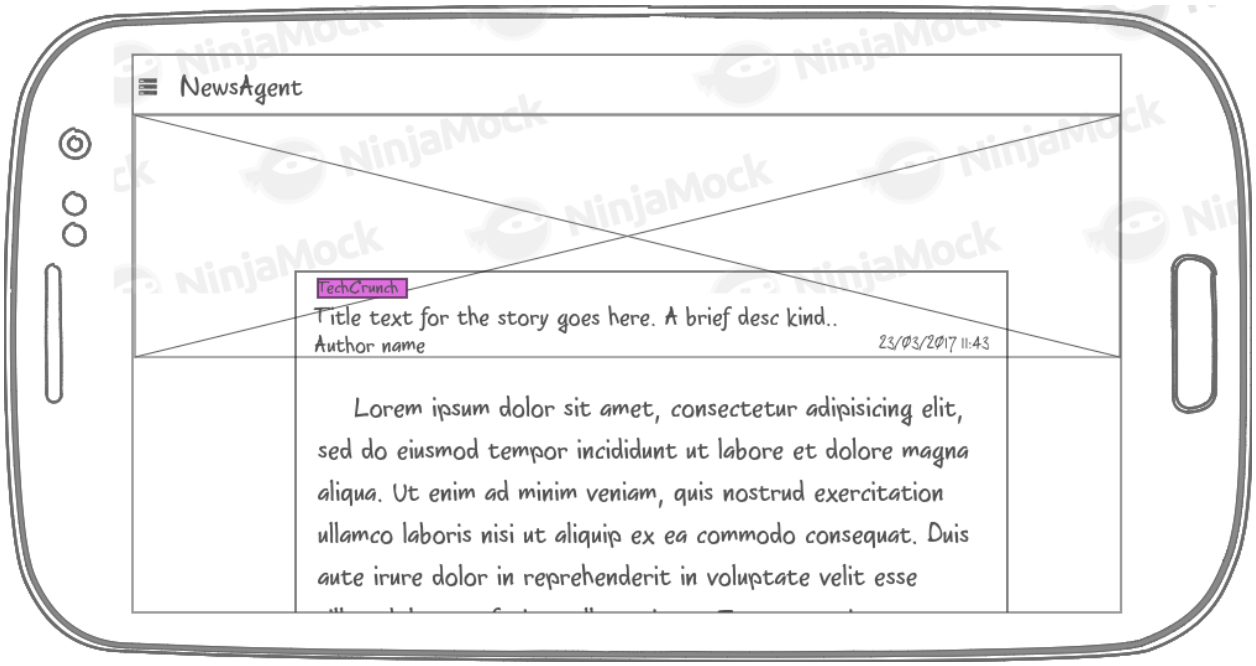

Categories selection screen. This screen can be accessed by using Navigation drawer menu selection.

## Screen 5




This is one more news feed detail screen. This screen do not have any tabs to swipe for categories. This screen will be accessed after a category selection.

## Screen 6




This screen is a feed details screen for larger screen sizes.

## Screen 7


This screen is a widget mockup screen. In this screen user can see the latest news feed and move back and forth using the right corner buttons.

# Key Considerations

### How will your app handle data persistence?

NewsAgent application will use content provider as data persistence option. NewsAgent application will only store the liked news feeds locally.

### Describe any edge or corner cases in the UX.

User can share the stories by using a share icon which will be displayed on the image in details screen.

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

Glide - To handle loading and caching the images .
Retrofit - This is type-safe HTTP client library which is useful to wrap around any WEB API call.
Butterknife - This is a view 'injection' library which is widely use in android application for binding field and method with android views using annotation.

# 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 Android project using Android studio.

- Set Minimum API Level as 14
- Configure libraries

## Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Setup Tabbar with Collapsed Toolbar Layout to start it from middle of the screen and settle down in the top
- Build UI for feed list items
- Build UI for categories item
- Build UI for feed details screen

## Task 3: Implement Data Layer

- Create database and helper classes
- Create content provider and setup Loader to update the Like page

## Task 4: API Integration

- Create helper classes to make API calls using retrofit
- Load the API data into UI

## Task 4: Final Steps

- Create free and paid flavor
- Wrap up UI with large screens
- Corner cases: Create Broadcastreceiver to detect network connectivity changes and display the warning message using Snackbar.
- Create a placeholder screen background which displays the internet connectivity status
- Create a placeholder screen background to display empty stories status.
- Add Firebase Jobdispatcher poll the api endpoint every 1 hour and pull latest news feeds.

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

---

**Submission Instructions**

- After you've completed all the sections, download this document as a PDF [ File → Download as PDF ]
    - Make sure the PDF is named "**Capstone_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"