

**GitHub Username:** NaturalFractals

# NewsReporter

## Description

Are you tired of getting your fingers dirty with ink while reading the newspaper, or maybe you just want easy access to the New York Times without having to waste all of that paper. This app will provide you with the most up-to-date news from the New York Times.

## Intended User

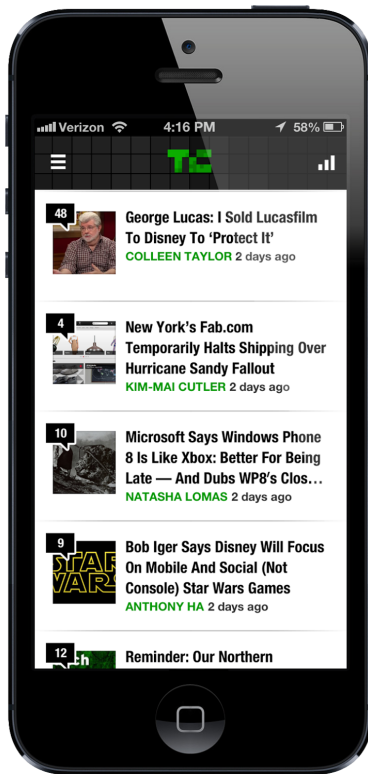
Family, friends, adults, children, and anyone who reads the newspaper.

## Features

- Keep up to date with the latest news from the New York Times
- Ability to save interesting articles to read for later.

## User Interface Mocks

### Screen 1



**\*\*NOTE:** This image is from google images

[https://www.google.com/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=0ahUKEwikylznu87NAhXIYiYKHR7DBjlQjRwIBw&url=http%3A%2F%2Fblog.aol.com%2Ftag%2Fapp%2F&bvm=bv.125801520,d.eWE&psig=AFQjCNFhFTyM6oMuv\\_pCmEcLk3MbOtibkw&ust=1467331998935961](https://www.google.com/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=0ahUKEwikylznu87NAhXIYiYKHR7DBjlQjRwIBw&url=http%3A%2F%2Fblog.aol.com%2Ftag%2Fapp%2F&bvm=bv.125801520,d.eWE&psig=AFQjCNFhFTyM6oMuv_pCmEcLk3MbOtibkw&ust=1467331998935961). This image provides a representation of the layout of my app.

This is the main screen of the app. It contains a ListView of the most up-to-date news from the New York Times. The list items will be clickable, clicking on a list item will go to a detail screen with the full news story. The Favorites section of the app will contain an identical layout.

## Screen 2



The detail screen will contain an image from the news story, the main title of the new story, and the full news story. There will be a Floating Action Button to add the story to the user's favorites. This will allow the user to view the story later, even if offline.

## Key Considerations

How will your app handle data persistence?

My app will use AsyncTask to load data from the web. When the user saves a news story the data from the news story will be saved in a SQLite Database. This news story will then be accessed through a Content Provider and displayed under the user's Favorites. This will allow the user to view the news story while offline.

Describe any corner cases in the UX.

The user can return to the main feed of news stories from the detail news screen through the back button. If the user is viewing a news story in their favorites the back button will take them back to the List of their Favorite news story, and then further back to the main feed of news story.

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

Retrofit - downloading and parsing JSON.

Picasso - downloading and caching images from the news story.

## 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

- Design project and setup packages, classes, and dependencies.
- For this project I will be using Retrofit and Picasso as third party libraries (can use any third party libraries that you like).
- Get API key from New York Times developer website, and read up on documentation.

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

- Will be using fragments inside of each activity for the UI setup. You will have at minimum a Main Activity with a fragment and a Detail Activity with a Fragment.

### Task 3: Implement AsyncTask for Fetching Data

- Use an AsyncTask to fetch data from the New York Times API. This is where I will use Retrofit, but there are many options for fetching data from an API.
- Two Async Task will likely be needed. One for fetching the main list of news articles, and another for fetching data on a specific article (Detail Activity).

### Task 4: Implementing Adapters for Displaying Data

- Implement Adapters for each of the Activities and their fragments. I will be using Picasso to help with uploading images into the Views, but you may use Glide or another method.

## Task 5: Implementing Data Persistence using SQLite Database

- A database will be needed to achieve data persistence in the application. Saving the information of a news article locally will allow the user to view the article while offline.
- Classes for the Database Contract, Helper, and Content provider will need to be created for this task.
- App must use a Loader to move data into the views.

## Task 6: Implement existing UI to work for tablet

- Up until this point the app has been designed to work on a mobile phone. We need to adjust the application so that it will be compatible and user-friendly on a tablet device. This can be done by using a two pane view on tablet devices.

---

### 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**"