

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Screen 3](#)

[Tablet UI](#)

[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.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Create Database and ContentProvider](#)

[Task 3: Implement UI for Each Activity and Fragment](#)

[Task 4: Optimize layouts for Tablet UX](#)

[Task 5: Add Google Analytics and Location Support](#)

GitHub Username: LadwaAditya

TwiTone

Description

This app is basically a Twitter Client for android. The main focus of this app is to provide access to user's timeline, mentions, direct messages offline. So data persistence is one of the major area that I will be focusing on.

Intended User

The app is intended for avid Twitter users

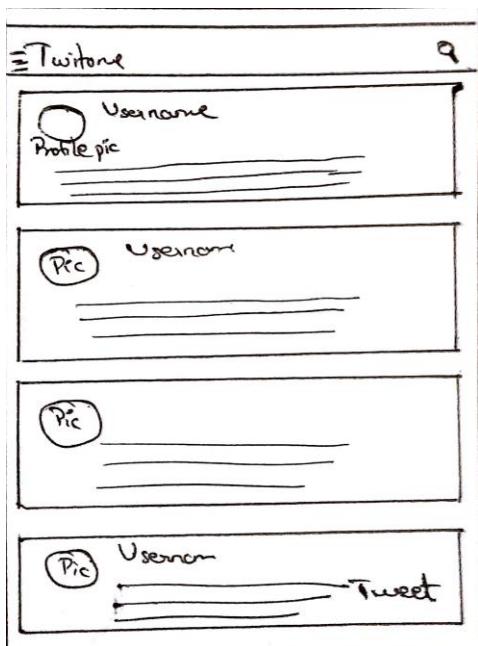
Features

- Post Tweets
- Access user's timeline, mentions, direct messages offline

- Material Design
- Access Global and Local Trends

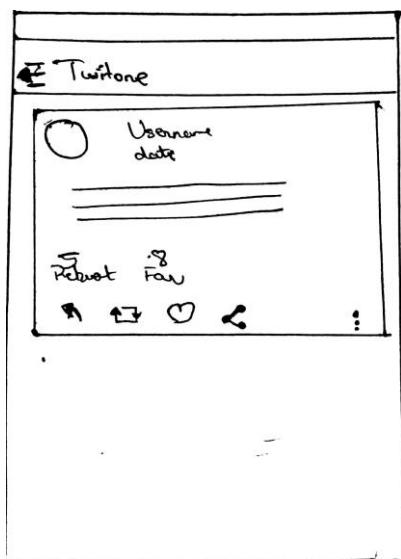
User Interface Mocks

Screen 1



In this screen the user can see his Timeline of Tweets. The tweet will be displayed on a CardView. It will also contain Profile pic, Username, Tweet and on clicking the card the detail of the tweet will be opened

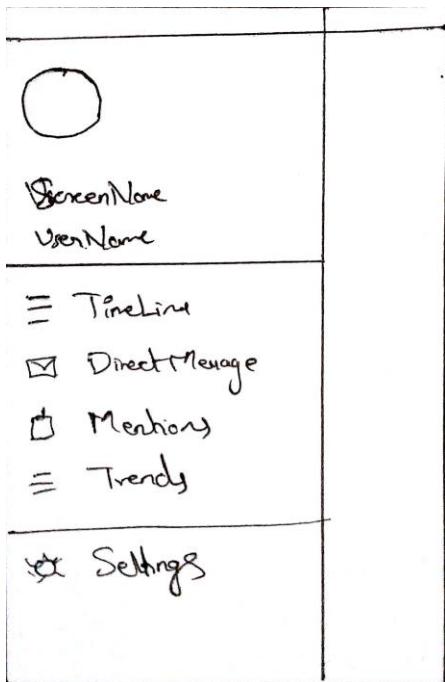
Screen 2



Capstone_Stage1

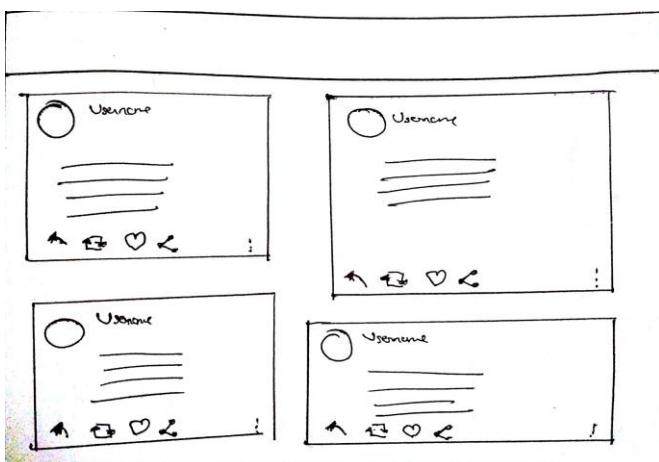
In detail Tweet screen the user can see the details of the screen. He can also Like, Retweet, Replay to Tweet and Share the Tweet.

Screen 2



In Navigation Drawer the user can navigate to their timeline, Directmessage and Messages. Users can also navigate to Trends and Setting of the app

Tablet UI



In Tablet UI I'm thinking to use a Staggered GridView but not sure if it will be good for users. This will utilize the extra space available for the user

Key Considerations

How will your app handle data persistence?

Twitter data will be accessed by making REST calls to Twitter API. Users data will be cached in a local database. A content provider will be used for efficient data persistent and accessing using Loaders.

Describe any corner cases in the UX.

Navigation Drawer will be used that allows users to navigate to their Timeline, Mentions and Direct messages

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

- AppCompat - Adds support for the Action Bar user interface design pattern
This library includes support for material design user interface implementations
- Design → Material design components to all developers and to all Android 2.1 or higher devices. You'll find a navigation drawer view, floating labels for editing text, a floating action button, Snackbar, tabs, and a motion and scroll framework to tie them together.
- Glide → Handling ImageView recycling and download cancelation in an adapter, complex image transformations with minimal memory use, and automatic memory and disk caching.
- Dagger 2 - Dependency Injection library

Next Steps: Required Tasks

Task 1: Project Setup

Get Twitter API keys and setup project Libraries

- We login to <https://dev.twitter.com/> and create an app and retrieve the API key and access token and secret

- Setup all the required libraries for the App

Task 2: Create Database and ContentProvider

Create Helper classes for implementing database and contentprovider

Task 3: Implement UI for Each Activity and Fragment

- Build UI for LoginActivity
- Build UI for MainScreenl

Task 4: Optimize layouts for Tablet UX

Optimize layouts for tablets

Task 5: Add Google Analytics and Location Support

- In this task I add analytics and Location support.
- Location will be used to post a tweet along with users current location