GitHub Username: anagnostou74

My Stories

Description

Tired to read news? Looking for inspired stories and things to laugh? Get in "My stories" application and your free time will change forever! Read interesting, funny stories every day and get inspired from famous quotes.

Intended User

The app is addressed to the average person who wants to read or see something funny and get undistressed from every day's pressure.

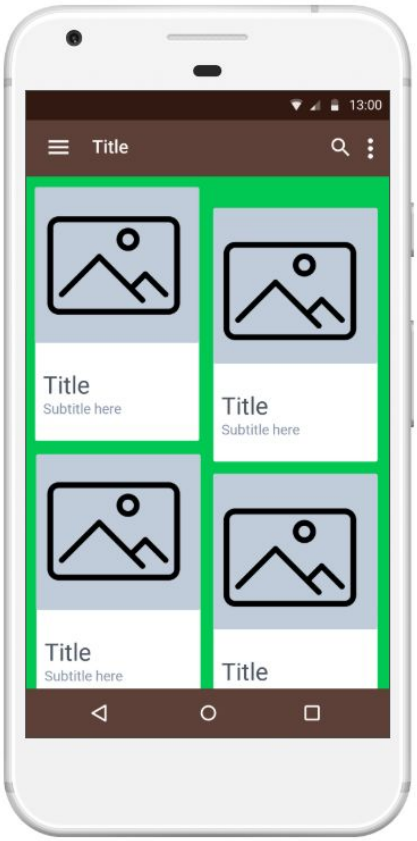
Features

A user will be capable to read stories, view pictures and see video. The content will be updated every day from a Firebase database.

User Interface Mocks

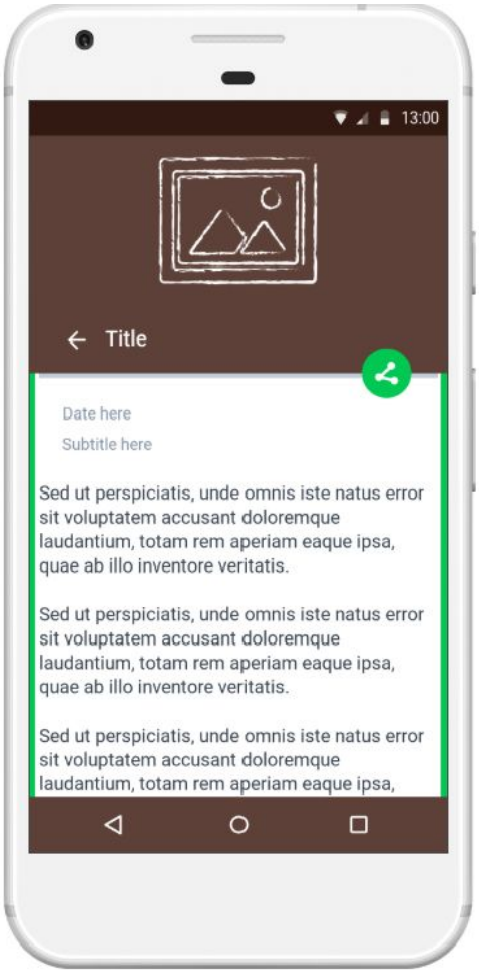
Screens designed with marvelapp.com

Screen 1



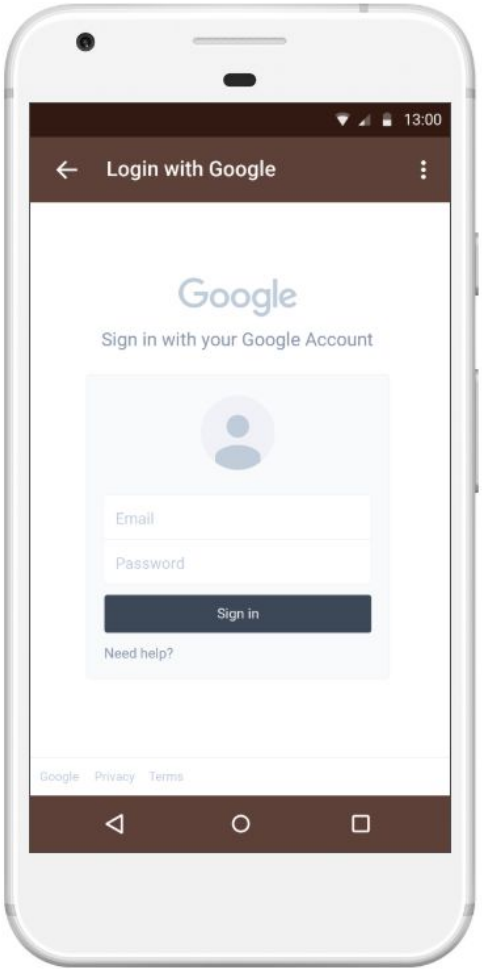
"My Stories" main screen with a GridLayoutView of the most current stories.

Screen 2



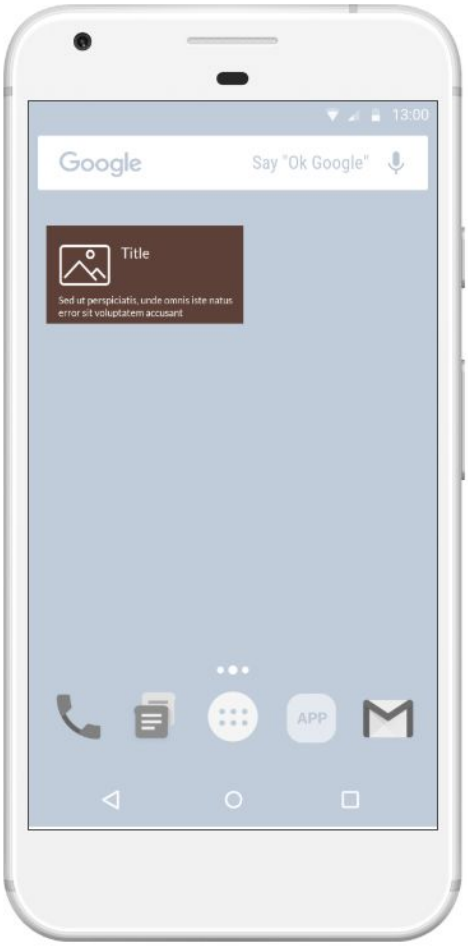
A single story detail view UI with share button.

Screen 3



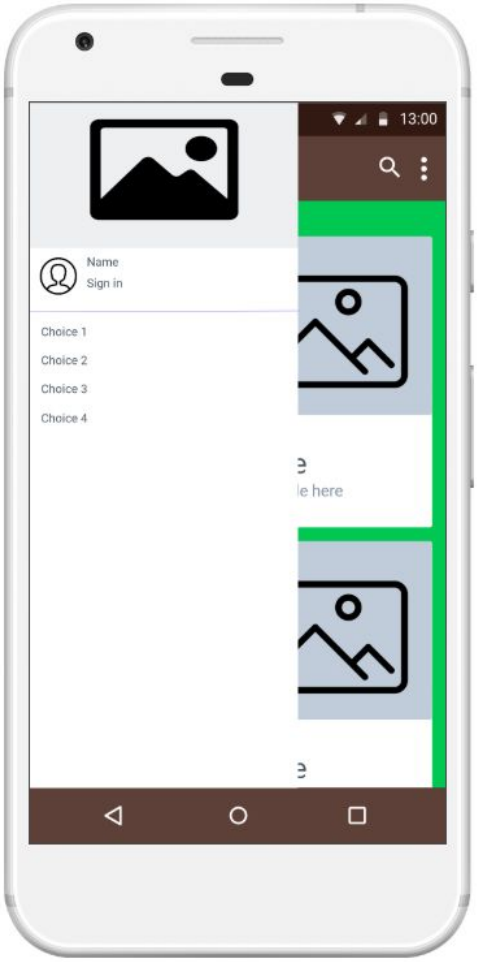
A sign in screen to let the user put his account credentials and keep his favorite stories.

Screen 4



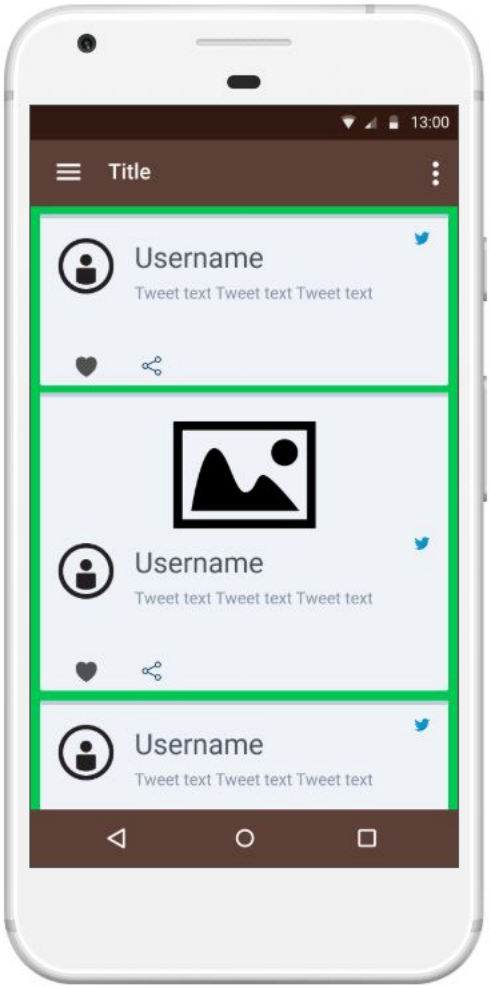
A home screen view with “My Stories” widget which drives the user in the displayed story.

Screen 5



A drawer layout with user choices to navigate through the app and sign in details.

Screen 6



A list view layout which brings all the latest tweets from a Twitter list.

Key Considerations

How will your app handle data persistence?

My Stories will use a Firebase Realtime Database to bring new content to users.

Describe any edge or corner cases in the UX.

A RecyclerView with cardview items will deliver the story's image and title. After a users selection and in a constraint layout the user will read the main story, will see (if it exists) the video and the image. The back button will terminate the current view (a video for example) and

will return to the previous one. If that view is the main list then the user will go back to his home screen.

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

- ButterKnife, for better UI and code engagement.
- Glide for image loading and caching.
- Exomedia to display videos.
- Design and cardview support libraries.
- Firebase instant messaging and database.
- Google analytics to understand what the user wants.
- Firebase authentication.
- OkHttp library for better network connections

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

- Firebase instant messaging will be used to implement push notifications.
- Firebase Realtime Database will be used to deliver the content.
- Google Firebase analytics will be used to track key events.
- Firebase authentication with Gmail will be used to give user the ability to save favorite content.
- Twitter API to fetch the funniest and interesting tweets from a specific list.

Required Tasks

Task 1: Project Setup

"My stories" application will be written solely in the Java Programming Language, with Android Studio IDE and the necessary libraries mentioned before. After the initial setup I will create a firebase database with demo content. A Git repository will be created to support the development.

Stable release versions of all libraries, Gradle, and Android Studio will be used.
Android Studio 3.1.3

```
dependencies {
   implementation fileTree(dir: 'libs', include: ['*.jar'])
   implementation 'com.android.support:appcompat-v7:27.1.1'
   implementation 'com.android.support:design:27.1.1'
   implementation 'com.android.support.constraint:constraint-layout:1.1.2'
   // Messaging
   implementation 'com.google.firebase:firebase-messaging:17.0.0'
```

```
    // Analytics
    implementation 'com.google.firebase:firebase-core:16.0.1'
    // Authentication via Google mail
    implementation 'com.google.firebase:firebase-auth:16.0.2'
    implementation 'com.google.android.gms:play-services-auth:15.0.1'
    // Realtime Database
    implementation 'com.google.firebase:firebase-database:16.0.1'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
    // Butterknife
    implementation 'com.jakewharton:butterknife:8.8.1'
    annotationProcessor 'com.jakewharton:butterknife-compiler:8.8.1'
    // Glide
    implementation 'com.github.bumptech.glide:glide:4.7.1'
    annotationProcessor 'com.github.bumptech.glide:compiler:4.7.1'
    // Exomedia
    implementation 'com.devbrackets.android:exomedia:4.2.1'
    // Twitter
    implementation 'com.twitter.sdk.android:twitter-core:3.3.0'
    implementation 'com.twitter.sdk.android:tweet-ui:3.3.0'
    // OkHttp
    implementation 'com.squareup.okhttp3:okhttp:3.10.0'
}

// Top level dependencies
dependencies {
    classpath 'com.android.tools.build:gradle:3.1.3'
    // NOTE: Do not place your application dependencies here; they belong
    // in the individual module build.gradle files
    classpath 'com.google.gms:google-services:4.0.1'
}
```

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for story's details
- Build UI for Drawer menu.

Task 3: Add Firebase implementation

- Firebase instant messaging.
- Google Firebase analytics.
- Firebase authentication with Gmail.

Task 3: Add Detail activity

- Build a Detail Activity with the necessary support libraries to display a text, an image (if supported) and a video (if supported).

Task 4: Add Twitter activity

- I will build an AsyncTask with the appropriate Activity, which will bring to the user the funniest tweets from a Twitter user's list.