

# Group 29 (AKT)

## Requirements Specification

Alex Trudeau  
400030148

Kathryn Kodama  
400013582

Tommy Tran  
001150067

December 6, 2017  
Version 2.2

# Contents

<b>1</b>	<b>Project Drivers</b>	<b>3</b>
1.1	Purpose of the Project . . . . .	3
1.2	Stakeholders . . . . .	3
<b>2</b>	<b>Project Constraints</b>	<b>4</b>
2.1	Mandated Constraints . . . . .	4
2.1.1	Solution Constraints . . . . .	4
2.1.2	Implementation Environment of the Current System . . . . .	4
2.1.3	Partner or Collaborative Applications . . . . .	4
2.1.4	Off-the-Shelf Software . . . . .	5
2.1.5	Schedule Constraints . . . . .	5
2.2	Naming Conventions and Terminology . . . . .	5
2.3	Relevant Facts and Assumptions . . . . .	5
<b>3</b>	<b>Functional Requirements</b>	<b>7</b>
3.1	The Scope of the Work . . . . .	7
3.1.1	The Current Situation . . . . .	7
3.2	Business Data Model & Data Dictionary . . . . .	8
3.3	The Scope of the Product . . . . .	9
3.4	Functional Requirements . . . . .	10
<b>4</b>	<b>Non-functional Requirements</b>	<b>13</b>
4.1	Look and Feel Requirements . . . . .	13
4.2	Usability and Humanity Requirements . . . . .	13
4.3	Performance Requirements . . . . .	13
4.3.1	Speed and Latency Requirements . . . . .	13
4.3.2	Reliability and Availability Requirements . . . . .	14
4.3.3	Longevity Requirements . . . . .	14
4.4	Operational and Environmental Requirements . . . . .	14
4.5	Maintainability and Support Requirements . . . . .	14
4.5.1	Maintenance Requirements . . . . .	14
4.6	Security Requirements . . . . .	15
4.7	Cultural Requirements . . . . .	15
4.8	Legal Requirements . . . . .	15
<b>5</b>	<b>Project Issues</b>	<b>16</b>
5.1	Open Issues . . . . .	16
5.2	Off-the-Shelf Solutions . . . . .	17
5.2.1	Ready-Made Products . . . . .	17
5.2.2	Products that can be copied . . . . .	17
5.3	New Problems . . . . .	17
5.3.1	Limitations in the Anticipated Implementation Environment That May Inhibit the New Product . . . . .	17
5.3.2	Follow-Up Problems . . . . .	17

5.4	Tasks . . . . .	18
5.4.1	Project Planning . . . . .	18
5.5	Risks . . . . .	18
5.6	Costs . . . . .	18
5.7	User Documentation and Training . . . . .	18
5.8	Waiting Room . . . . .	19
5.9	Ideas for Solutions . . . . .	19

## List of Tables

1	<b>Revision History</b> . . . . .	2
2	<b>Terminology</b> . . . . .	5
3	<b>Users schema</b> . . . . .	8
4	<b>Subreddits Schema</b> . . . . .	8
5	<b>Posts Schema</b> . . . . .	9
6	<b>Comments Schema</b> . . . . .	9
7	<b>Project Planning</b> . . . . .	18

## List of Figures

1	Implementation Environment Diagram . . . . .	4
2	Current Situation Diagram . . . . .	7
3	ER Diagram . . . . .	8
4	Use Case Diagram for the User Interaction System . . . . .	10

Table 1: **Revision History**

Version	Date	Notes
1.0	04/10/17	Created document
1.1	05/10/17	Updated document
1.2	06/10/17	Finalized document Rev. 0
2.0	26/11/17	Started Rev.1
2.1	29/11/17	Updated document
2.2	06/12/17	Finalized document

# 1 Project Drivers

## 1.1 Purpose of the Project

The purpose of the project is to create a community based social news aggregator and message board. Users are able to rate posts based on their interest. Posts are then sorted from highest to lowest rated, essentially allowing for more interesting topics to be shown to a greater audience. In addition to viewing posts, users will be able to comment and discuss within each post. This will facilitate the creation of and discussion within online communities.

## 1.2 Stakeholders

There are multiple stakeholders for this application as listed below.

### The Customer/End Users

- **Application Registered Users:** The primary customers of this project are the application registered users. These are members of the general public who have an authenticated profile within the application and post and update shared content.
- **General Public:** The secondary customers of this project are the general public users who browse the content without contributing and curating content.

### The Principals and Partners

- **AKT Team:** The AKT Team are the developers of this project. The team is involved in the creation and maintenance of the project. In this case, the developers are the principal stakeholders as well as the producers.
- **Google Firebase:** Other producer stakeholders including Google Firebase as they are hosting the application and providing user authentication and database services.

## 2 Project Constraints

### 2.1 Mandated Constraints

#### 2.1.1 Solution Constraints

**Description:** The application shall operate on devices modern web browsers.

**Rationale:** The users include anyone with a device that can operate a web browser.

**Fit Criterion:** The application shall be tested to be functional on web browsers across several devices.

**Description:** The application shall operate only when the device is connected to the internet.

**Rationale:** In order to submit content or participate in discussions, users require an internet connection.

**Fit Criterion:** The application shall not load content without an internet connection.

#### 2.1.2 Implementation Environment of the Current System

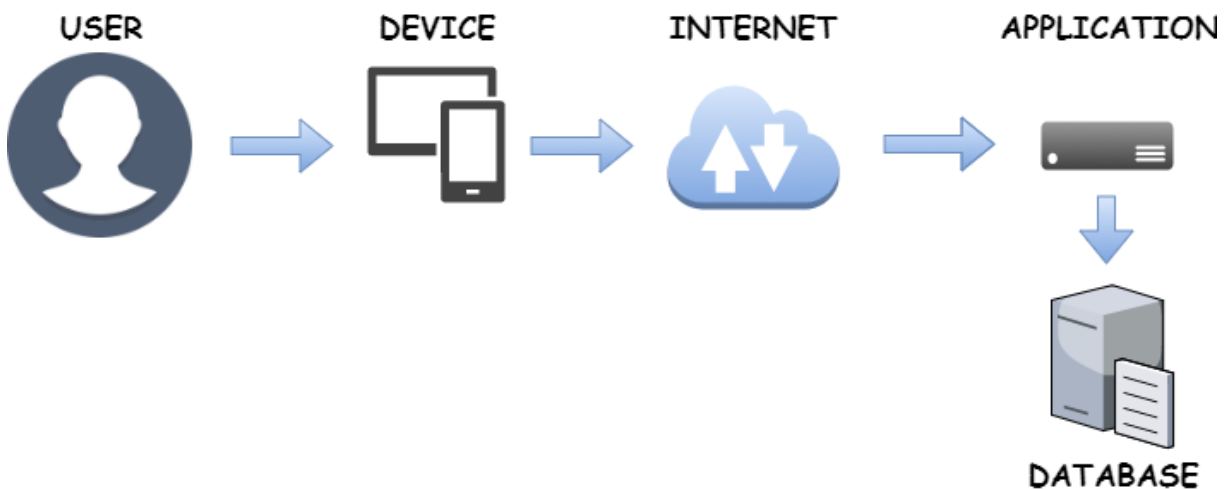


Figure 1: Implementation Environment Diagram

#### 2.1.3 Partner or Collaborative Applications

The application will leverage Google Firebase's services in order to handle user authentication, database communication and web server hosting.

### 2.1.4 Off-the-Shelf Software

The application requires that the user have a device that is able to connect to the internet and operate a web browser(Chrome, Firefox, Safari, Edge, Opera, etc).

### 2.1.5 Schedule Constraints

The proof of concept demonstration application must be completed by October 16, 2017.

The application must be completed by December 6, 2017. The development team can choose to maintain and update the project at their convenience after this date.

## 2.2 Naming Conventions and Terminology

The naming conventions of this project have been agreed upon by the AKT development team. Lower case Camel-Case will be closely followed when writing code. Additionally, constants and enums will be identified with all capital naming conventions. Each page and component of the code will follow standard ionic project naming conventions. For example, the homepage of the project will be named as follows: home.html, home.module.ts, home.scss, home.ts.

Table 2: Terminology

Term	Description
AKT Team	The development team
Comment	A user's response to a post or another user
Down-vote	A way for a user to 'vote-down' a comment or post
Firebase	A service offered online that hosts applications, provides user authentication and databasing services
GUI	Graphical User Interface
Post	A submission by a user that contains text or a link
Reddit	The open source project being mimicked, can be viewed on <a href="https://www.reddit.com/">https://www.reddit.com/</a>
Reddit-Clone	The current project being designed
Subreddit	Page holding the posts of a certain topic
Subscribe	A function for users to "favourite" certain subreddits.
Unsubscribe	A function for users to remove a subreddit from their list of subscribed subreddits.
Up-vote	A way for a user to 'vote-up' a comment or post
User	A person using the application

## 2.3 Relevant Facts and Assumptions

The original application is written in Python, an object oriented class-based programming language. Since the AKT team is developing this project using JavaScript (still an object oriented language but not class-based) there is an assumption that all basic functionality can

still be mimicked without relying on the same class-based system.

This application is intended for use by the general public. It is assumed that users will not require more than very basic knowledge of their device and web browser in order to access the application. It is assumed that users of this application will have internet connectivity in order to access the application.

If a user deletes their profile, the content they shared and contributed under that profile will remain on the application.

## 3 Functional Requirements

### 3.1 The Scope of the Work

The scope of this project is to mimic the basic functionality of Reddit with a new design. This core functions of this project are user authentication, creation of threads, comments, and the ability to rate content through up-votes and down-votes. The project will be constructed using Node.js and JavaScript which any device with web browsing capabilities will be able to access.

#### 3.1.1 The Current Situation

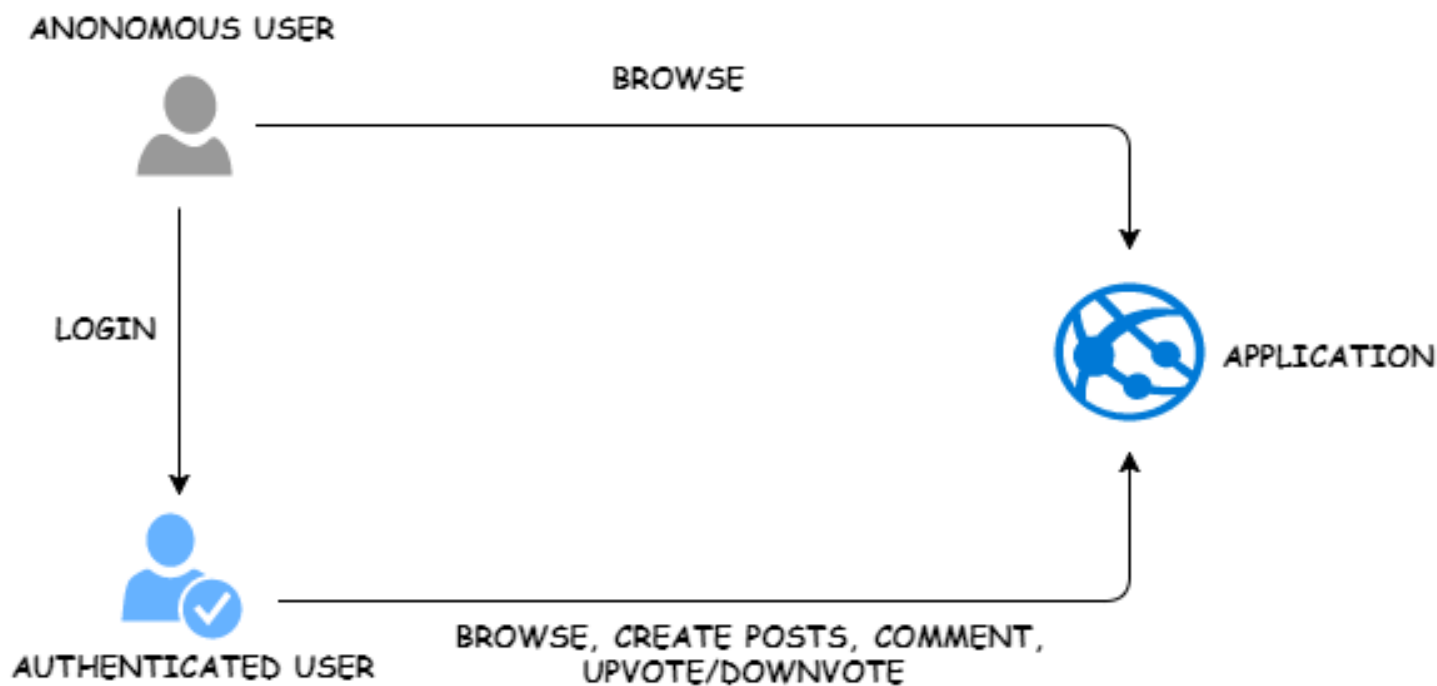


Figure 2: Current Situation Diagram



### 3.2 Business Data Model & Data Dictionary

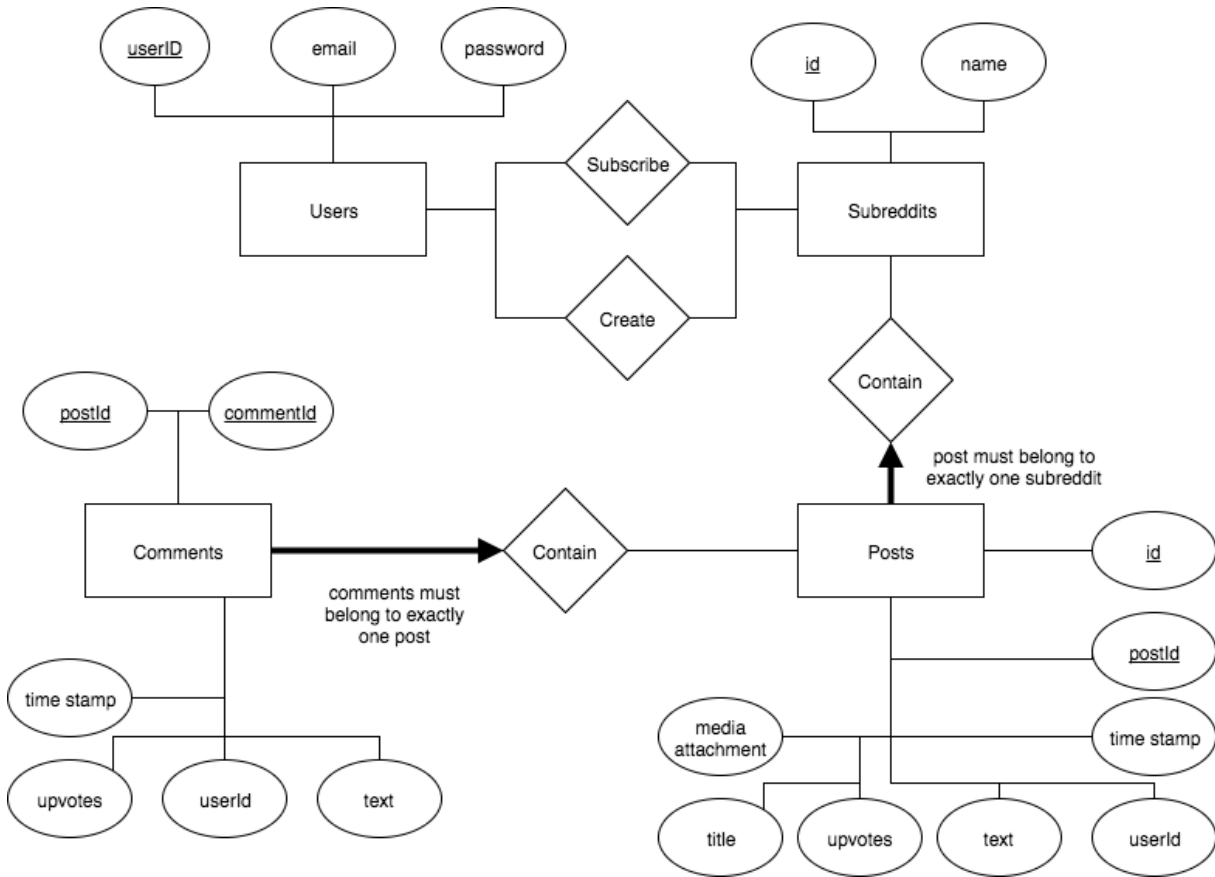


Figure 3: ER Diagram

Table 3: Users schema

Parameter	Type	Description
uid	String	Firebase assigned user id.
email	String	User's specified email address
password	String	User's specified user password. Password combined with email address allows users to access their account.

Table 4: Subreddits Schema

Parameter	Type	Description
subredditId	String	Unique identifier of a subreddit.
name	String	Name of the subreddit.

Table 5: **Posts Schema**

Parameter	Type	Description
postId	String	Unique identifier auto-assigned to post.
subredditId	String	id Referenced from Subreddit table that ties a post to a subreddit.
uid	String	userId referenced from Users table that ties a user to a post.
media_attachment	String	URL or binary data (uploaded from device)
title	String	User specified title of the post
message	String	Text that belongs in the content of a post.
up-votes	Integer	Number modified as user's 'up-vote' a post or 'down-vote' a post.
time_stamp	Number	UNIX timestamp of the date at time of creation.

Table 6: **Comments Schema**

Parameter	Type	Description
commentId	String	Unique identifier auto-assigned to comment.
postId	String	postId referenced from Posts table that ties a comment to a post.
uid	String	userId referenced from Users table that ties a user to a comment.
upvotes	Integer	Number modified as user's 'up-vote' a post or 'down-vote' a comment.
message	String	Content of the comment.
time_stamp	Number	UNIX timestamp of the date at time of creation.

### 3.3 The Scope of the Product

Below is the Use-Case Diagram for the User Interaction System.

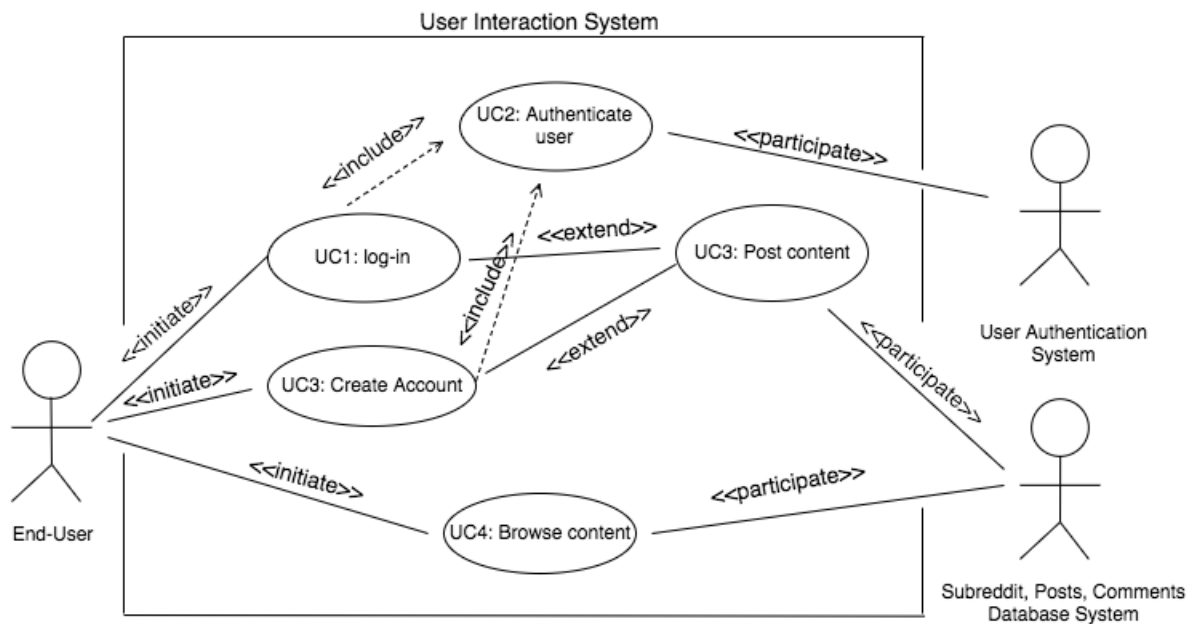


Figure 4: Use Case Diagram for the User Interaction System

### 3.4 Functional Requirements

**Requirement Number: FR1**

**Description:** A user should be able to create a profile, log-in and log-out at their will.

**Rationale:** If the user cannot create a profile, they cannot create posts or contribute content. Similarly, a user should have the right to permanently remove themselves from the application through deletion of their profile.

**Fit Criterion:** Able to create a user profile, verify through an email address, reset one's password, and delete application.

**Requirement Number: FR2**

**Description:** A user who is logged in should be able to create a post within a subreddit.

**Rationale:** Any authenticated user should be able to post content.

**Fit Criterion:** Posts shall be created on multiple accounts.

**Requirement Number: FR3**

**Description:** Any user should be able to browse content (not required to log-in). This includes navigating to subreddits and viewing corresponding posts and comments.

**Rationale:** Users should not need to be authenticated when just browsing the application.

**Fit Criterion:** Both authenticated and anonymous users shall be able to browse through posts and comments in any subreddit.

**Requirement Number: FR4**

**Description:** Users should be able to delete their posts and comments when logged on.

**Rationale:** Users should have control over the content and replies they submit.

**Fit Criterion:** Authenticated users shall have the ability to edit and delete their posts.

**Requirement Number: FR5**

**Description:** A user should be able to create a comment in response to a post when logged on.

**Rationale:** Users should be able to discuss content posted.

**Fit Criterion:** Authenticated users shall be able to comment on posts.

**Requirement Number: FR6**

**Description:** A user should be able to 'up-vote' or 'down-vote' posts and comments when logged on.

**Rationale:** Users should be able to have input on what they deem is interesting and be able to agree or disagree with other user's comments. This will ensure the content on the application is user-curated.

**Fit Criterion:** Authenticated users shall be able to 'up-vote' and 'down-vote' posts or comments once.

**Requirement Number: FR7**

**Description:** Posts can be sorted based on date posted and popularity.

**Rationale:** More popular or interesting posts(deemed by the community) should be shown to users with priority.

**Fit Criterion:** Posts with higher ratings shall be shown before lower rated posts.

**Requirement Number: FR8**

**Description:** User profiles show karma points(accumulated from their posts or comments)

**Rationale:** Users submitting quality content or have comments which the community values should be awarded.

**Fit Criterion:** User profiles shall show their accumulated karma points from all of their posts and comments.

**Requirement Number: FR9**

**Description:** Users should be able to search for their desired subreddits.

**Rationale:** There can be a large a number of subreddits as the number of users increase. Having the option to search for subreddits will make the end user experience more enjoyable..

**Fit Criterion:** There shall be a search bar at the top of the home page where users can search for a subreddit that matches their input.

**Requirement Number: FR10**

**Description:** Users should be able to subscribe to their favourite subreddits as well as unsubscribe.

**Rationale:** There can be a large number of subreddits on the application as the number of users increase. Users being able to favourite subreddits will help them navigate to their desired subreddits faster.

**Fit Criterion:** When viewing a subreddit users will be able to click a "subscribe" button. The subscribed subreddit will show up when they enter the search bar on the home page. Similarly, when viewing a subreddit users are subscribed too they will be able to click an "unsubscribe" button.

**Requirement Number: FR11**

**Description:** Users should be able to create subreddits.

**Rationale:** Users should be able to create new subreddits so that they can facilitate new discussions and communities.

**Fit Criterion:** When on the home page in the top right menu users should be able to select "Create a subreddit" where they are prompted to add the title and description.

## 4 Non-functional Requirements

### 4.1 Look and Feel Requirements

**Requirement Number:** NF1

**Description:** The GUI will be how the user interacts with the application. The GUI should include a log-in/sign-up screen, content posted on the application and click-able buttons. The buttons will allow the user to navigate the application as well as 'up-vote' and 'down-vote' content. Content should be sorted based on popularity.

**Fit Criterion:** The user shall able to interact with the application's interface easily. A sample of people will test out the application and confirm ease of use.

### 4.2 Usability and Humanity Requirements

**Requirement Number:** NF2

**Description:** The application will be able to be used by anyone that can use a keyboard and mouse, or a mobile device's keyboard and UI.

**Fit Criterion:** Users with both mobile or desktop devices shall be able to interact with the application.

### 4.3 Performance Requirements

#### 4.3.1 Speed and Latency Requirements

**Requirement Number:** NF3

**Description:** The application shall respond to user input to make subreddits, posts, and comments. The application data shall update with new subreddits, posts, comments as soon as the user opens the application, or manually refreshing the page.

**Fit Criterion:** Newly added posts,subreddits and comments shall be shown after navigation, content submittal or a page refresh.

### 4.3.2 Reliability and Availability Requirements

**Requirement Number:** NF4

**Description:** The application shall be available wherever there exists an internet connection and as long as the Firebase realtime database is under maximum load.

**Fit Criterion:** The page shall only load when on a device with an internet connection and there are under 100 users using the application.

### 4.3.3 Longevity Requirements

**Requirement Number:** NF5

**Description:** The application will be relevant for the life of Angular 4 and Ionic 2/3, as well as Javascript functional browsers.

**Fit Criterion:** The application shall compile locally with Angular version 4.x.x and Ionic version 3.x.x.

## 4.4 Operational and Environmental Requirements

**Requirement Number:** NF6

**Description:** The product is to be used anywhere where there exists an internet connection, and there exists a device that has access to an internet browser.

**Fit Criterion:** The application shall be functional on devices with a working internet browser and internet connection.

## 4.5 Maintainability and Support Requirements

### 4.5.1 Maintenance Requirements

**Requirement Number:** NF7

**Description:** The application will have minimal maintenance requirements. The domain name will have to be renewed yearly. Changes to visual styling may be made without affecting the functionality of the application itself.

**Fit Criterion:** The application shall remain active without without downtime for an indefinite amount of time.

## 4.6 Security Requirements

**Requirement Number:** NF8

**Description:** Each user profile is password protected. This protects others from logging into another user profile and posting/updated/deleting content that is not theirs. There is an option for user's to reset their password as well via a "Forgot your password?" button.

**Fit Criterion:** Users shall be required to authenticate themselves before viewing their user profiles.

## 4.7 Cultural Requirements

Non-applicable.

## 4.8 Legal Requirements

**Requirement Number:** NF9

**Description:** The application should adhere to all laws

**Fit Criterion:** All licenses for libraries used shall be licensed for free use



## 5 Project Issues

### 5.1 Open Issues

Issues can be viewed on the reddit-clone [GitHub](#).

1. **Back end database management; Closed as of October 1, 2017**

Authentication through Firebase has been completed but how the application manages data storing and data flow is still an issue.

2. **Posts can be created even when user is not logged in; Closed as of October 14, 2017**

When selecting "create a post" for the second time and the user does not exist or is not logged in, they will still be able to create a post.

Opened by: Kathryn Kodama

Closed by: Tommy Tran

3. **Browser issue; Closed as of October 21, 2017** Application is not fully compatible with ES7 code (may want to switch to ES6).

Opened by: Alex Trudeau

Closed by: Alex Trudeau

4. **Mobile login not working on Android Google Chrome; Closed as of November 17, 2017** When on mobile (Android) and trying to log in, when incorrect password or email entered cannot edit the entered fields.

Opened by: Kathryn Kodama

Closed by: Kathryn Kodama, Alex Trudeau - new version of Ionic seemed to fix this problem.

5. **Verification email inconsistency; Closed as of November 26, 2017**

Users should be able have resend a verification email in case the original for some reason is not received.

Opened by: Tommy Tran

Closed by: Kathryn Kodama

6. **Fresh clones of repository do not compile user metadata; Closed as of November 19, 2017**

When the Github repository is cloned an error occurs on the user metadata.

Opened by: Kathryn Kodama

Closed by: Tommy Tran, Alex Trudeau

**7. String .valueOf() does not work in some browsers; Closed as of November 25, 2017**

Opened by: Alex Trudeau

Closed by: Alex Trudeau

**8. User's subscribed subreddits not syncing with other devices; Closed as of November 25, 2017**

When a user unsubscribes from the mobile login it does not appear as unsubscribed on another device.

Opened by: Alex Trudeau

Closed by: Alex Trudeau

## **5.2 Off-the-Shelf Solutions**

### **5.2.1 Ready-Made Products**

Since the application requires unique data management, it is not possible to purchase a solution.

### **5.2.2 Products that can be copied**

The application is based off of the original Reddit so it is possible to observe the source code and try to mimic their solution.

## **5.3 New Problems**

### **5.3.1 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product**

Since the application is leveraging Firebase's free plan, if there are over 100 simultaneous connections or data transferred is over the allowed limit the application may not function correctly.

### **5.3.2 Follow-Up Problems**

The application requires Firebase in order to function properly. If Firebase is shut down or offline there is nothing our developers can do about it. Updates of Firebase can subsequently alter the compilation of the project and therefore the project must be updated accordingly.

## 5.4 Tasks

### 5.4.1 Project Planning

Table 7: **Project Planning**

Task	Projected Finish Date
User authentication	October 14, 2017
Subreddit, Post and Comment functionality	October 16, 2017
User profiles and subreddit/post sorting	November 13, 2017
Additional functionalities	November 27, 2017

A more detailed breakdown of the project can be viewed via the Gantt chart on the reddit-clone [GitHub](#)

## 5.5 Risks

As with any online open forum or discussion board, a health and safety risk associated with this project is cyber-bullying or online harassment. An additional health and safety risk associated with this application is strain on the eyes as it only operated through the screens of mobile devices or user's computers. There is also the case of copyright infringement or inappropriate content being posted. This will be up to the developers to prevent and curate while this application is in use. The developers email will also be available for users to report specific cases of harassment or inappropriate use of the application.

Risks to be tested for the proof of concept are user security. It is critical to test that the project will protect the email and passwords of its' users. Additionally, the proof of concept should test for FR1, FR2, FR3, FR5, FR6. The remaining functional requirements will be implemented and tested after the initial proof of concept. The appropriate corresponding non-functional requirements will be tested as the functional requirements are tested.

## 5.6 Costs

There will be no external costs involved in the implementation of the project because of the free hosting service provided by Google Firebase. However, if there are too many users or posts a premium Firebase plan may need to be purchased.

## 5.7 User Documentation and Training

The project will include a "Help" page which will instruct users upon creation of a log-in on how to create subreddits, posts, and upvote content. The GUI will be easy enough to navigate that documentation will not be needed for users in order to browse content.

## 5.8 Waiting Room

The following requirements are not to be included in the initial product release.

- Curating and moderating content including but not limited to appointing moderators. This would require additional developers.
- Additional functionality including new sorting methods for content.
- Restriction of inappropriate content including an option for users to report other users, comments, posts, and subreddits.
- A messaging systems for users to have one-on-one conversations.

For the first proof of concept the following functionalities were ready to roll out: user authentication, creating posts, subreddits, comments and up-voting and down-voting content.

Beyond the proof of concept (for the final demo) the following functionalities were added: user profile, sorting posts and subreddits, reddit-clone themes and deleting comments and posts.

## 5.9 Ideas for Solutions

Further insight on Firebase User authentication and security. Proper hierarchy, teamwork, and documentation of code with help to solve issues as they occur.