**Nevermore**

**Software Requirement Specification (version 1.1)**



**Fall 2020**

Software Project Management Plan

Project Manager: Paul Gibbs

Team Members: Ulises Santana

Christian Strickland

Emanuel Carbajal

Daniel A. Mendez

Hussain Alabdrabalnabi

Jose Duarte

**Table of Contents**

1. Introduction……………………………………………………………………………………...4

1.1 Purpose……………………………………………………………………………….4

1.2 Scope…………………………………………………………….……………………4

1.3 Definitions, Acronyms, and Abbreviations………………………….4

1.4 References………………………………………………………………………....5

1.5 Overview…………………………………………………………………………….5

2. Overall Description……………………………………………………………………………5

2.1 Product Perspectives………………………………………………………....5

2.1.1 System interfaces (deployment diagram) ..............5

2.1.2 User interfaces……………………………………………………6

2.1.3 Software interfaces…………………………………………….6

2.1.4 Communication interfaces………………………………….6

2.1.5 Memory……………………………………………………………..6

2.1.6 Operation…………………………………………………………...6

2.2 Product Functions (use case diagram) ..................................7

2.3 User Characteristics…………………………………………………………...

2.4 Constraints…………………………………………………………………………

2.5 Assumptions and Dependencies…………………………………………

3. Specific Requirements………………………………………………………………………

3.1 External interface requirements…………………………………………

3.1.1 User interfaces…………………………………………………...

3.1.2 Hardware interfaces……………………………………………

3.1.3 Software interfaces…………………………………………….

3.1.4 Communication interfaces………………………………….

3.2 Functional requirements……………………………………………………

3.3 Performance requirements……………………………………………….

3.4 Design constraints…………………………………………………………….

3.5 Software system attributes………………………………………………

3.5.1 Reliability…………………………………………………………..

3.5.2 Maintainability………………………………………………….

3.5.3 Portability………………………………………………………...

1. **Introduction**

**1.1 Purpose**

The following information in this document will explain in detail the functionality, use cases, frameworks and plans for the Nevermore project. This document is intended for management, technical, and structural purposes.

The Nevermore project is a Twitter API based web application that allows for users to set a time limit for their social media posts called tweets on Twitter. It is hosted on its own site which will allows users to login and then create the tweet and allowing a timer to start until it is to be destroyed. An addition of a timer will allow for users to increase their follower interactions as well as allow for user to send temporary messages. By building upon Twitter and their API we are able to expand the usability of the platform hopefully bringing a new innovative way for Twitter users to use the social media platform. The UI and feature set will change over time as we gather new ideas.

**1.2 Scope**

The application will provide an interface to be able to create tweets that will be removed after a timer countdown ends. Furthermore, the application will provide a way for end users to discover new content in our Explore section by browsing tweets that use our hashtag #nevermore

As other social platforms do, our project is expected to have multiple users and does not aim to have a specific audience. The planning of our project will include a web application that is also optimized for a mobile browser. For feedback, we expect to have sample clients and from our CEO and instructor Bilal Khan.

**1.3 Definitions, acronyms and Abbreviations**

CSE 4550: Computer Science and Engineering – Software Engineering 4550

CSS: A style sheet language used to describe visuals of a website

CSUSB: California state university, San Bernardino

Google Cloud Platform: A proposed backend platform for our project

HTML: Hypertext Markup Language, a system for tagging font, color and graphics inside a web page

JS: JavaScript, a programming language used for interactive effects in a web browser

SRS: Software requirement specification

UI: User interface, the interfaces and objects being interacted with by the user.

Web app: A program or application built for a device running an internet browser.

Web browser: An app used for browsing the web. The main browsers are Chromium based (Google Chrome, Microsoft Edge etc.), Safari and Mozilla Firefox.

**1.4 References**

[1] *Twitter,* [https://twitter.com](https://snapshat.com)

[2] *Snapchat,* [https://snapchat.com](https://snapshat.com)

**1.5 Overview**

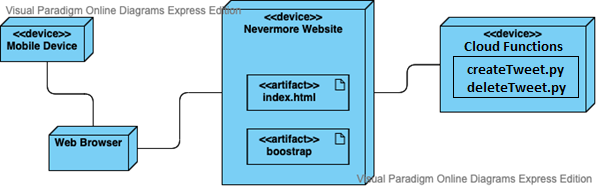
The following information in this document will explain in detail the functionality, use cases, frameworks and plans for the Nevermore project.

Section 2 of this document will describe the user interfaces, technologies and constraints that make up the project.

Section 3 of this document will describe the requirements needed for the project. This includes specific features, functionality and performance requirements.

The Nevermore project is a web application optimized for desktop and mobile devices that will aim to connect and socialize with verified system users. It will aim to become a subjectively better version of existing social media platforms. It is proposed that the project will be hosted on Amazon web Services or Google Cloud platform. The UI and features will change over time as we gather new ideas.

1. **Overall Description**
   1. **Product Perspectives**
      1. **System Interface (Deployment Diagram)**



Major components involved in this project include the website and its database which will be accessed by a web browser of the users choice from their mobile device.

* + 1. **User Interfaces**

When a user navigates to the Nevermore website, they will either first be prompted with an introduction page which will give a short detailed view of this website and prompt the user to either sign up or log in (new user/not logged in) or return to the home screen (logged in). The home screen includes the Nevermore logo along with user function buttons to return to home screen, access account details, access direct messages, view following/followers. From the home screen the user may view their feed which contains screeches from the users followed and the time they will expire unless re-screeched.

* + 1. **Software Interfaces**
       1. **Traditional Web Application** (Contains all code which makes up Nevermore)
          1. HTML
          2. JS
          3. CSS
          4. Bootstrap
       2. **Cloud Functions**
          1. Python
    2. **Communication Interfaces**

This will be accessed between the web browser from the mobile device to the web server.

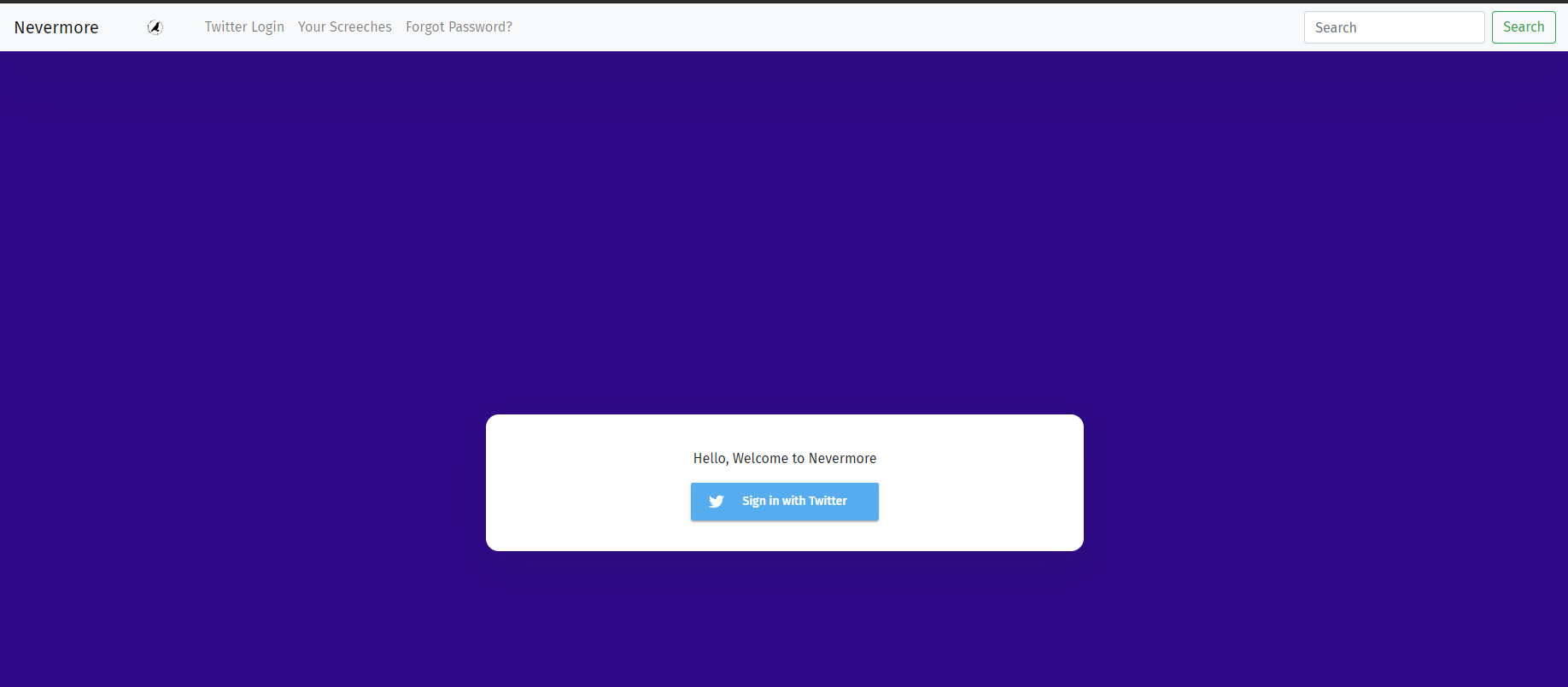
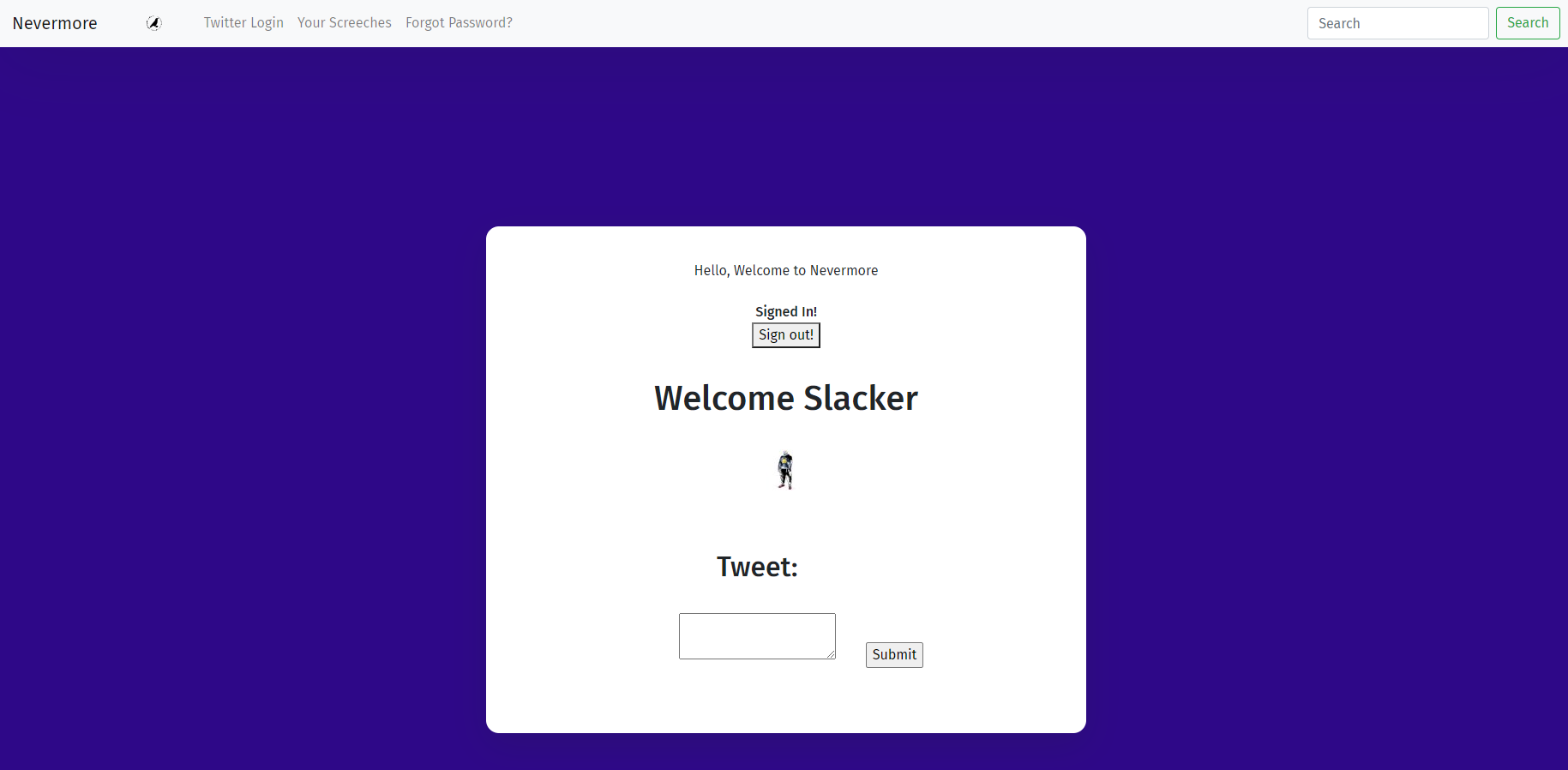
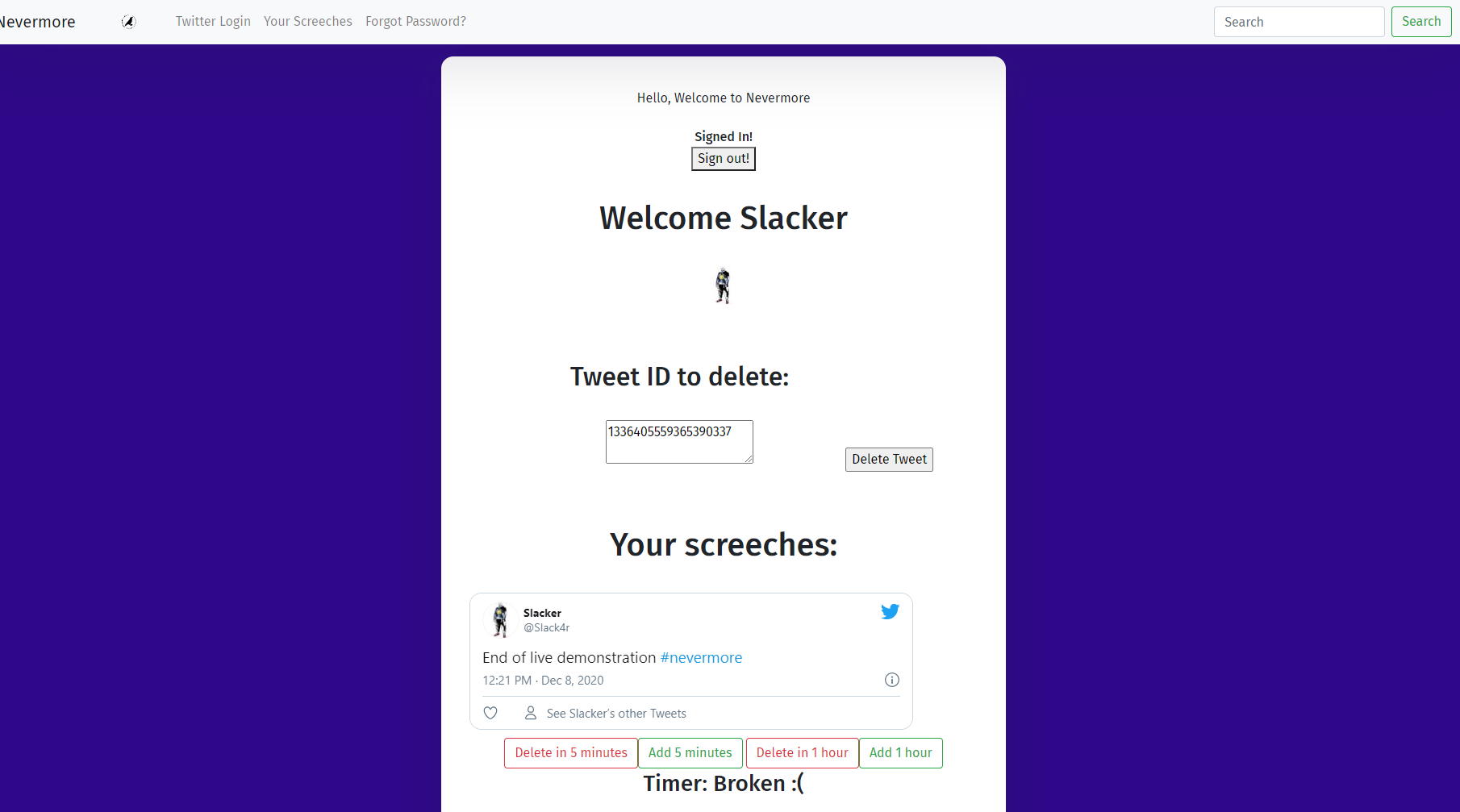
* + 1. **Memory**

Only limited to the amount of memory the dedicated running machine has which presently can be more than 256GB since for deployment all project files will be saved locally. When live, we will be required to choose a server that best fits the applications space requirement.

* + 1. **Operation**

Operation is hosted through firebase hosting as a part of Google Cloud Platform. Hosting is scalable so for current implementation it is free but if demand were to rise it would scale to meet the demand.

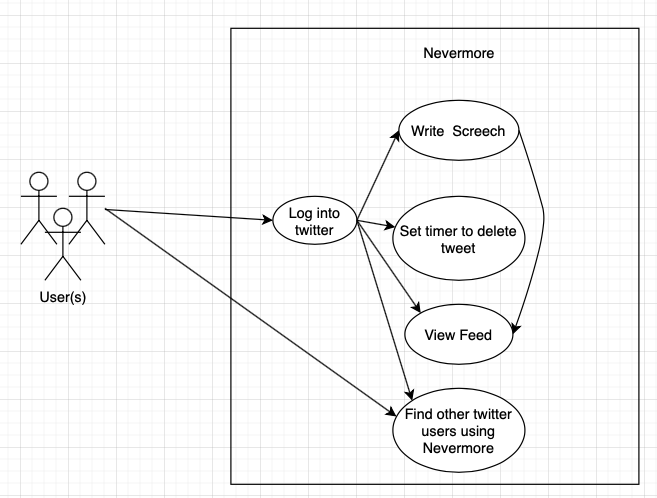
* + 1. **UI Examples**

Home screenLogged in Tweet creation pageYour screeched page

**2.2 Product Functions**

The application allows users to login through the Twitter API, create tweets, and manage their tweets all within our website.

Use case diagram:



**2.3 User Characteristics**

Users will be able to send direct messages to other uses, who can then reply if they choose to. Users can also view posts from other users as well as their own, which other users and themselves can interact with, such as by posting a comment. Users can choose to follow other users, and those users can choose to follow back. Users can have their own home menu where they will be able to view posts from other users they follow and can choose to interact with their content there.

**2.4 Constraints**

1. Learning to configure the server, setup the web application and database
2. Using Firebase if unfamiliar with it

**2.5 Assumptions and Dependencies**

1. Requires internet access
2. Requires stable functionality at all times

**3. Specific Requirements**

**3.1 External Interface requirements**

**3.1.1 User Interfaces**

The user interface consists of graphical elements in the website, this will include:

**Home page:** The main page of the website that will contain a user’s feed of screeches from the accounts that they follow along with ads and suggested content. UI will include a search bar, Home page button, hashtag feed, direct messages (whispers), User page, and a screech button

**Login Page:** Page that takes in a user email/username/phone number and password to allow them to login.

**Search Page:** Show results from what a user has searched as well as recommendations for similar topics or accounts

**Hashtag/Trending Page:** Show the popular trends for regional and global markets.

**Whispers Page:**  Container page for private conversations between users. These messages do not have an expiration time like normal screech messages

**User page:**  Page containing info about the user including following, followers, bio, profile picture, banner, screeches, echoes, and likes.

**Screech button:** Click to create new screech interface, controlling who can view the screech, later on what media to include, emotes, and initial time length for the screech.

**3.1.2 Hardware Interfaces**

There are no external hardware interfaces for this project.

**3.1.3 Software Interfaces**

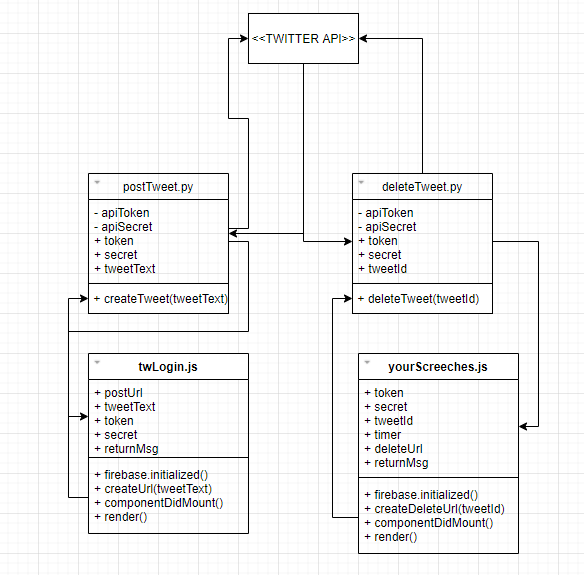
Nevermore communicates with its server on Google Cloud platform to retrieve and save information for the user. It loads all the posts that other users have made on the website and communicates with the server to request web page information for further data.

**3.1.4 Communication Interfaces**

Nevermore communicates with your computer or mobile device in order to send and receive user’s information and posts made on the webpage.

**3.2 Functional requirements and User Interfaces**

If a user is not logged into the webpage the login interface will be the first to load. If a user does not have an account, they may click on create account which will take them to create an account interface. If the user is already logged in you can see all the posts made from other users that you are currently following and see some trending posts as well.



Class diagram

**3.3 Performance requirements**

The Nevermore server must update user information in real time and update all other users posts in real time so that you won't miss any new posts made by other users.

The UI must be quick and responsive for the user to be more engaged and active on other users' posts.

The user must have a capable device to run a web browser and a modern web app. Most devices made in the last 10 years will perform well.

The web browser must be updated recently to support the latest web frameworks. Newer web browsers such as Google Chrome, Safari, and Mozilla Firefox should perform well. Older web browsers such as Internet Explorer are not guaranteed to perform well.

**3.4 Design constraints**

Both mobile and webapps must be connected to the internet in order to access database and retrieve user information. Without internet access application will not be able to retrieve data required to function properly.

**3.5 Software System attributes**

**3.5.1 Reliability**

Measures will be taken in order to ensure the reliability of the application and that the user data is secure. This means ensuring that only those who are authorized to view user's information will be able to do so.

**3.5.2 Maintainability**

Software will be sufficiently documented such that the application will be easy to maintain and modify. Future features which could be implemented will be kept in mind.

**3.5.3 Portability**

Current app cannot be ported to another language at this time.