Software Requirements Specification

for

Spyke

**Version 1.0 approved**

**Prepared by Ryan Smith, Duy Le,  
Andrew Brewer, Wolfgang de Groot**

**CSE 4214, Group 8**

**2022-01-25**

# Introduction

## Purpose

The purpose of this SRS is to describe the general outline of our project. This paper includes general details, goals, and development plans for this project.

## Document Conventions

**Bold text** is used to identify priorities and important steps during the process of creating this software, as well as outlining sections used to identify what the reader is reading.

## Intended Audience and Reading

This document is targeted towards Developers, Project managers, and Documentation Writers. The rest of the document contains information on the functionality, scope, and organization of the software application. For Project Managers, it is recommended reading this paper in order. For Developers and Documentation Writers, it is recommended that you read 1.2, 1.4, and then focus on sections 2-4 for a list of goals to achieve.

## Product Scope

The product is intended for use by the public as a means for an easier method of communication amongst family, friends, and colleagues. Benefits of this product include promoting the freedom of speech, creating new business opportunities, and promoting international communication for the public. Current objectives include personalized accounts, a public posting feature, a like / dislike system, and a sharing system. Future goals include a direct messaging system, and a timeline system to properly date events. With the implementation of the timeline feature, like / dislike system, and sharing system, we can sell accurate, dated information to third parties and provide targeted ads to our users.

## References

There are no references to style guides, contracts, or system requirements to date. This section will be updated once these are drafted and published.

# Overall Description

## Product Perspective

This product is a new, self-contained product. This is the first iteration of this product line and is not building on top of any prior project.

## Product Functions

* Users can create an account.
* Users can log into their account.
* Users have a timeline made up of statuses.
  + Users can compose, edit, post their own statuses.
  + Users can share friends’ status on their own timeline.
  + Users can determine if their status is public or private.
* Users have a friends list.
  + Users can send friend requests and add friends if accepted.
  + Users can accept a friend request.
  + Users can delete already accepted friends from their friends list.
* Users can view others’ timeline, depending on friendship status and public/private status. Users can like and comment on friends’ timeline.

## User Classes and Characteristics

The privileges are constant throughout all classes. These classes are defined by their use of the platform and their impact on potential future monetization prospects.

Class 1: The teen addicted to social media

Importance: most important class

Technical prowess: limited

Frequency of use: deleteriously high

Educational attainment: highschool/college

Experience: many according to their timeline

Functions used: all of them

General notes: this user type provides 80% of engagement, even though they are 20% of

our users. They provide most of our data which we sell to third parties for profit. They do not

read the terms of service and have no idea we do this, the perfect mark.

Class 2: The lurker

Frequency of use: high

Functions used: friend requests, viewing others’ timelines

General notes: this user does not contribute to the conversation and provides very little data

for us to sell to third parties. Targeted advertising is also difficult due to the lack of data.

This user is still important as they provide network analysis data. Through network analysis

we can determine roughly what the user’s interests are for targeted advertising.

## Operating Environment

2.4.1 Hardware Platform

We will be hosting the web application through the Google Cloud Platform (GCP) using a distribution of Ubuntu as the operating system.

2.4.2 Software Components

Linux

Apache Web Server

PHP

Composer, Packagist

SQL

## Design and Implementation Constraints

Due to the fact that this application is not particularly computationally intensive, there are no hardware constraints that arise. With respect to regulatory requirements, there are no constraints so long as the application acts as a platform and not a publisher. For corporate policies, this is an independent venture, therefore there are no additional requirements imposed by a potential customer. This app will be built using php, javascript, html, css, and SQL. The codebase will be maintained by our own developers.

# System Features

## Account Management and Control

3.1.1 Description and Priority

A method to create an account, login, logout, and delete that account freely.

This account will be used to authenticate an identity, interact with the platform by creating and sharing posts, and interacting with other users freely.

***Priority: HIGHEST***

3.1.2 Stimulus/Response Sequences

Upon creation of an account, the system will ask for a username and password. Once a valid query is given, the system will create the new user’s account and store its information in the database.

When logging in, the user will provide their username and password. Once a correct combination is given, the user is authenticated and given a session cookie.

When logging out, the session cookie is destroyed.

3.1.3 Functional Requirements

*For account management as a whole, a database must be accessible to the system at all times.*

***When creating an account:***

* *Username/Password requirements:*
  + *Must be able to conform to security requirements (4.3)*
* *When storing in database:*
  + *A new entry for the user is created*
  + *The password is hashed before storage!  
    Never ever store passwords in plain text!*
  + *All fields are in UTF-8, regardless of character limitations or requirements.*
  + *A unique ID will be assigned to the user for internal purposes only.*

*Errors*

ACC-1: Account Creation

ACC-2: Security Requirements Conformance

ACC-3: Database Control

ACC-4: Account Management

## Timeline Posting and Commenting

3.2.1 Description and Priority

*A method to allow users to create new posts and updates onto their timeline.*

***Priority: HIGH***

3.2.2 Stimulus/Response Sequences

On the front page of the website, a text box will be provided for the user to create a new post. Once it has been finished, that post will be appended to that user’s timeline.

Any user may comment on any other user’s posts.

3.2.3 Functional Requirements

*A database must be accessible to the system at all times.*

***When creating a new timeline post or comment:***

* *Prerequisite requirements:*
  + *A user must be logged in.*
  + *A user must have the privilege to create posts.*
* *Content requirements:*
  + *Between 1-500 characters*
  + *Must not violate site rules and guidelines*
  + *Must be sanitized of XSS and any other exploits*
  + *Any files attached to messages will be scanned with the VirusTotal API to detect malware*
* *When storing in database:*
  + *Encoded in UTF-8*
  + *Values:*
    - String*: postContent*
    - String*: author*
    - String*: uuid*
    - Int*: likeCount*
    - Int*: dislikeCount*
    - Int*: shareCount*
    - *Bool: isEdited*
  + *A unique ID will be assigned to the user for internal purposes only.*

POST-0: Database Connections

POST-1: Validating New Posts

POST-2: Creating New Posts

POST-X: Malware Detection

## Post and Comment Editing

3.3.1 Description and Priority

*The ability for users to edit their own posts and comments.*

***Priority: LOW***

3.3.2 Stimulus/Response Sequences

*For any given user, they will be shown the option to edit their posts.*

*Once edited, the post or comment will show an indication that it has been edited for transparency.*

3.3.3 Functional Requirements

*The database must be accessible to the system at all times.*

***When editing a post or comment***

* *Prerequisite requirements:*
  + *A user must be logged in.*
  + *A user must have the privilege to edit posts.*
  + *A user must only be able to edit their own posts.*
* *All content must be restricted to the same guidelines as creating a new one.*
* *Once edited, the* isEdited *bool in the database will permanently be set to true.*

EDIT-1: Editing posts

EDIT-2: Editing comments

3.4 **Friends**

3.4.1 Description and Priority

*The ability for users to ‘friend’ other users and share each other’s posts in each other’s timelines.*

***Priority: MEDIUM***

3.4.2 Stimulus/Response Sequences

*For any given user, their profile will include the ability to request to add them to the user’s friend list.*

*If accepted, each user will have the other’s profile appear as a friend.*

*The friended’s posts will appear on the friendee’s front page.*

*So long as it remains mutual, both users may post onto each other’s wall.*

3.4.3 Functional Requirements

*The database must be accessible to the system at all times.*

***When interacting with friends***

* *Prerequisite requirements:*
  + *A user must be logged in.*
  + *A user must have the privilege to interact with users.*
  + *A user must not be blocked by said user.*
* *State:*
  + *If the user isn’t friends with the other, the option to ‘****Add Friend’*** *will show.*
  + *If the user is friends with the other, the option to ‘****Remove Friend****’ will show.*
  + *If the user is blocked by the other, no options will be available.*
* *Mutual friends will be able to post onto each other’s pages/timeline. These posts will appear on the homepage of those who follow that user, even if the original poster is not mutual with them.*
* *Each user has a list of friends in the database as a list of unique IDs.*

NET-1: Adding Users as Friends

NET-2: Removing Users as Friends

NET-3: Posting to Friends User Walls

NET-4: Blocking Users

# Other Nonfunctional Requirements

## Performance Requirements

***Computer-***

Any personal computer (Windows, MacOS, most Linux distro) with access to the internet and a web browser (Mozilla Firefox, Safari, Edge)

***Phone and tablet computers-***

Modern phones and tablets (iPhones,iPad, Android phones and tablets, Windows phones?) with access to their respectable web browser along with access to the internet.

***Internet Access-***

Without access to the internet the user could not connect to the website at all.

## Safety Requirements

***Subject to the same cybersecurity requirements/laws of similar social networking sites, and must abide by the ethical guidelines set by the IEEE and ACM.***Must secure privacy of all users and monitoring of any suspicious activity performed with the use of the web application.

Staff moderation?

## Security Requirements

***Unique username-***

The usernames will be between 3-16 characters. Not already taken by any existing account. Allow a way of differentiating between different users on the platform.

***Password Security-***

User’s passwords must be between 8-128 characters must use a secure password

A mix of letters, numbers and symbols. Is not in the top most commonly used passwords

Passwords will be encrypted through a hash function to protect the user's account and personal information.

***Website authentication*** from the user and the server will be handled by the Web hosting provider.

Captcha to ensure there are no bots on the webpage to reduce unnecessary traffic.

***Main security requirement is the preservation of user identity. No third party should have access to user’s login information unless a breach of information occurs.***

## Software Quality Attributes

**1.Ease of use**

**2.Ease of access**

**3.Availability**

**4.Maintainability**

**5.Interpolation and testing**

# Other Requirements

At the time of writing this document there have not been any other requirements, but if there are any additional requirements it would be updated along the way.

**Appendix A: Glossary**

**IEEE-Institute of Electrical and Electronics Engineers**

**Professional association**

**ACM -Association for Computing Machinery**

**Appendix B: Analysis Models**

At the time of writing this document there have not been any models used, but if there are any additional changes it would be updated along the way.

**Appendix C: To Be Determined List**

IEEE Guidelines - https://www.ieee.org/about/corporate/governance/p7-8.html

ACM Guidelines - https://ethics.acm.org/