|  |
| --- |
| BSc (Hons) in Computing – Year 4 – Software Development |
| Requirements Specification (RS) |
| Social Modifications |

|  |
| --- |
| Keith Feeney  15015556  keith.feeney@student.ncirl.ie  November 2018 |

Requirements Specification (RS)

Document Control

Revision History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Version** | **Scope of Activity** | **Prepared** | **Reviewed** | **Approved** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Distribution List

|  |  |  |
| --- | --- | --- |
| **Name** | **Title** | **Version** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Related Documents

|  |  |
| --- | --- |
| **Title** | **Comments** |
| Title of Use Case Model |  |
| Title of Use Case Description |  |

**Table of Contents**

[1 Introduction 4](#_Toc316977392)

[1.1 Purpose 4](#_Toc316977393)

[1.2 Project Scope 4](#_Toc316977394)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc316977395)

[2 User Requirements Definition 4](#_Toc316977396)

[3 Requirements Specification 4](#_Toc316977397)

[3.1 Functional requirements 4](#_Toc316977398)

[3.1.1 Use Case Diagram 5](#_Toc316977399)

[3.1.2 Requirement 1 <name of requirement in a few words> 5](#_Toc316977400)

[3.1.3 Requirement 2 <name of requirement in a few words> 6](#_Toc316977401)

[3.2 Non-Functional Requirements 7](#_Toc316977402)

[3.2.1 Performance/Response time requirement 8](#_Toc316977403)

[3.2.2 Availability requirement 8](#_Toc316977404)

[3.2.3 Recover requirement 8](#_Toc316977405)

[3.2.4 Robustness requirement 8](#_Toc316977406)

[3.2.5 Security requirement 8](#_Toc316977407)

[3.2.6 Reliability requirement 8](#_Toc316977408)

[3.2.7 Maintainability requirement 8](#_Toc316977409)

[3.2.8 Portability requirement 8](#_Toc316977410)

[3.2.9 Extendibility requirement 8](#_Toc316977411)

[3.2.10 Reusability requirement 8](#_Toc316977412)

[3.2.11 Resource utilization requirement 8](#_Toc316977413)

[4 GUI 8](#_Toc316977414)

[5 System Architecture 8](#_Toc316977415)

[6 System evolution 8](#_Toc316977416)

# Introduction

## Purpose

The purpose of this document is to set out the requirements for the development of Social Modifications, a social networking web application and Android app designed for people with tattoos, piercings and other body modifications. As this is a social networking site, Social Modifications will be available to people over the age of 18 only. A chat bot will also be included in the web application and Android app as a help point for various aspects of the social network or just general conversation.

## Project Scope

Social Modifications (SM) is a web application and Android app. It is designed to be a social networking platform for people with, or have an interest in tattoos, piercings and modifications.

SM is a safe space for users to communicate with other users and share their body modification experience. The social network also has a chat bot. This chat bot is SM’s alternative to a help section.

With the likes of Facebook, Twitter, Instagram and WhatsApp, these have lengthy Help Sections that can be difficult to navigate through, and even then, the answer may not be clear. With the chat bot, users can communicate directly to it and get assistance from within the social network. The chat bot also can communicate events, such as the likes of Tattoo Conventions. It can also look up information on tattoo and piercings shops at the user’s request.

The web application and Android app will be copies of each other and they will both need Internet connection. But there will be a cached version available, so users can still have limited interaction with the social network, even if Internet problems exist.

Users will need to create an account with email and password, or via Google’s sign-in API, as only registered users can use the social network. Google Firebase is used for storage and encryption. System information is stored in Google Firebase Database, while the likes of images will be stored in Google’ Firebase Storage. The Android app will also need to search for accounts, so any linked Google account can be used to create an account on SM.

## Definitions, Acronyms, and Abbreviations

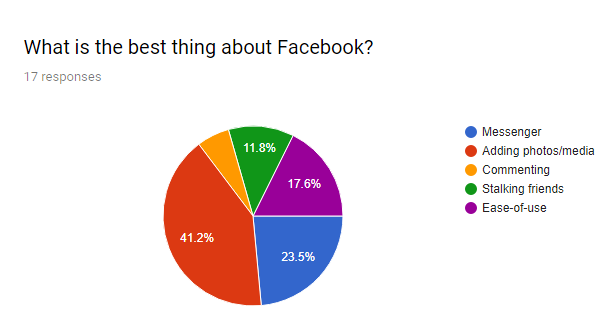
|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| SM | Social Modifications – The working name of the application |
| SN | Social Network |
| User | An end-user that interacts with SM. |
| Firebase Authentication | Used to verify users as well as handle the register / login |
| Firebase Database | Used to store data in a database (e.g. SQL info) |
| Firebase Storage | Used to store data but not in a database. (e.g. media) |
| AES256 | An encryption method to encrypt passwords. |
| GUI | Graphical User Interface |
| API | Application Programming Interface |
| Modified / Mods | To have tattoos, piercings or other body modifications |

# User Requirements Definition

Having communicated with modified friends about the idea of social networking application designed for them, one major thing kept appearing; simplicity. Many of my friends have said that current social networks have too much content, and most of it is just filler. There’s nothing valuable on them anymore.

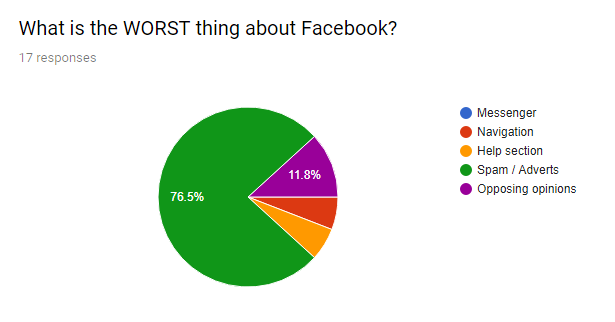
I created a form on Google Forms to see what people thought about social networking in general. As the form was being posted in Facebook Groups and Pages, I thought it would be best to stick with Facebook, as I could guarantee the participants know Facebook to some extent.

The first question was asking what is the best thing about Facebook.



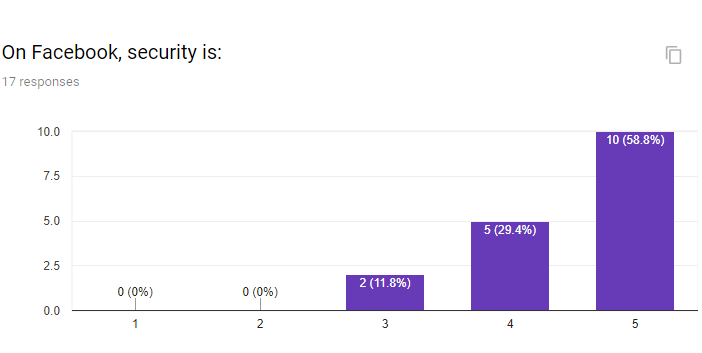
With this, most people decided that adding photos and media was the best thing about Facebook, with Messenger coming in second. Very few people think commenting is important. With Social Modifications, users need to be able to upload media and well as communicate with each other, as well as look up other users. Commenting doesn’t seem as important. This may be a future feature.

The second question asked was is the worst thing about Facebook.



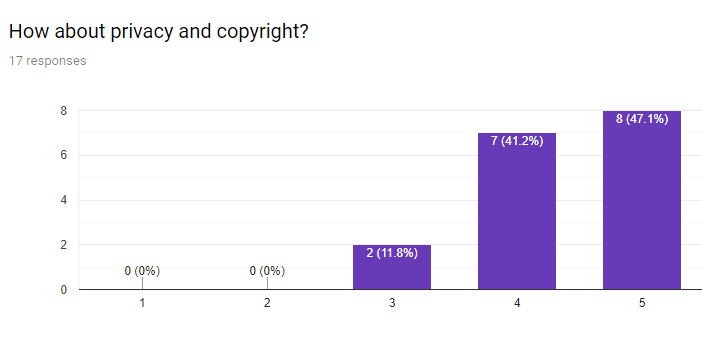
No surprise that ‘Spam/Adverts’ is deemed the worst thing about Facebook. But surprisingly, ‘Opposing opinions’ took the second majority. In SM, how adverts are handled needs to be seriously considered and not take away from the experience of the SN.

The third question looked at Facebook security and how important it is.



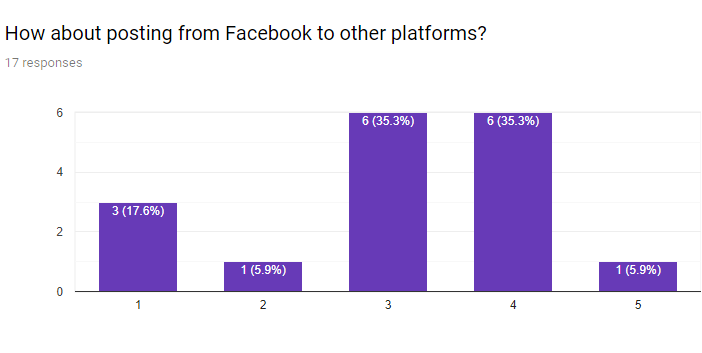
Not a shocker, but security is a major factor for the majority of people. It’s clear that in Social Modifications that data is transmitted securely and that it’s encrypted.

The fourth question looked at Facebook privacy and copyright.



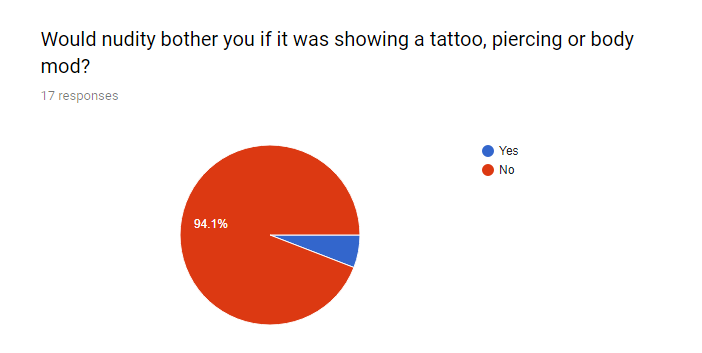
Privacy and copyright are important factors, but they seem to be slightly less important than security. In SM, private albums will exist so that users can share certain photos with other users, if they wish to. Another feature is that user’s will be able to put a watermark on their images if they want to.

The fifth question looked at sharing from Facebook to other platforms.



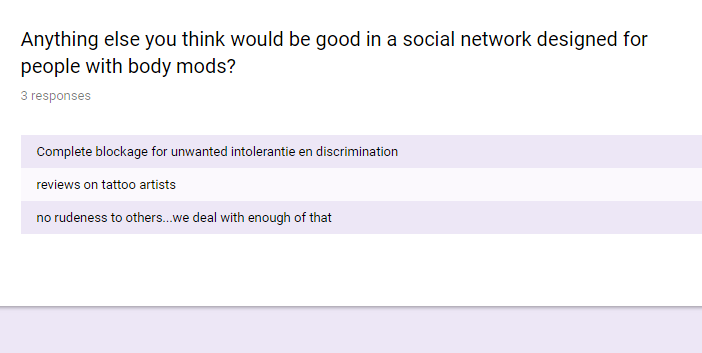
More people have decided that comparing to security and copyright, sharing to other platforms is not required as much. In SM, this could be added in a future update.

The sixth question added whether nudity would bother people I felt this was an important question as from talking with modified friends, modifications and nudity is not a sexual thing, but it’s more like art.



The majority of people agree with what I thought. This also confirms for me that Social Modifications should be for people aged 18 and older.

The final question was an open-ended question, so that people could add what they would like to see in SM.



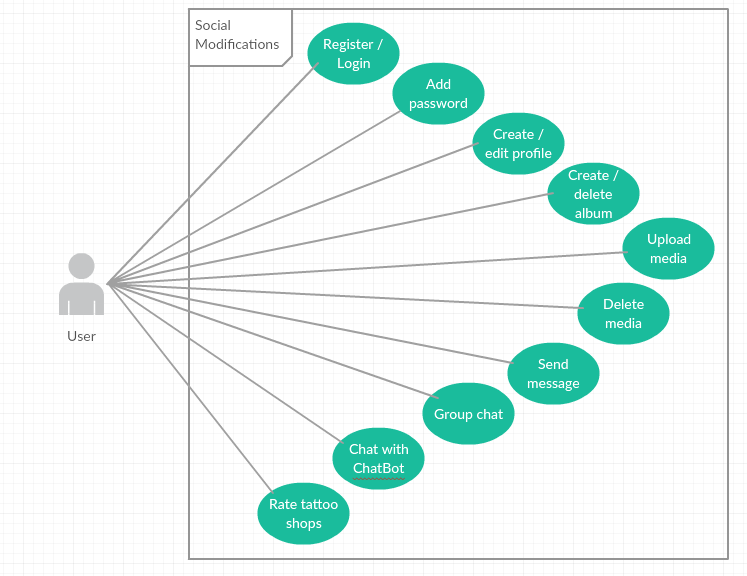
In SM, approval for new profiles, media, etc. seems to be important and will be implemented. The reviews on tattoo artists is a good idea, but I feel that would need to come from the likes of Google Maps, or Yelp or a similar source. I don’t think it would be beneficial to create a mini-rating site within SM on its initial release. Maybe on a future release.

# Requirements Specification

## Functional requirements

This section lists the functional requirements in ranked order. Functional requirements describe the estimated outcomes of the system.

### Use Case Diagram



.

### Requirement 1: Register/Login

#### Description & Priority

This is a very important requirement. Without this, users will not be able to access SM at all. Users will be able to register an account and if they have one already, they can login. User’s details will be authenticated by Firebase Authentication and it will be held in a Firebase Database.

#### Use Case

**Unique ID:** regLogin

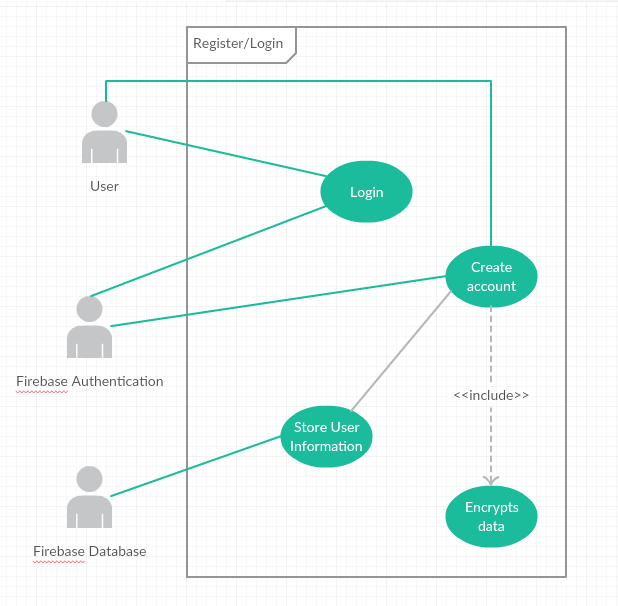
**Scope**

The scope of this use case is to show how the user interacts with the system when creating an account or logging into the system.

**Description**

This use case describes the user creating an account. If the user has previously created an account, they will skip registration and go to the login instead.

**Use Case Diagram**



**Flow Description**

**Precondition**

* The system is active and is in a wait state for a user.
* It is assumed the system is active and working correctly.
* The Login screen is shown to the user.
* A registration link is displayed on the screen.
* A Facebook login buttons is also available, so users can login using an external account.
* A token and unique user ID is created on Facebook and on SM to verify the user.
* No passwords are stored on the database, instead, an encrypted version of it is.

**Activation**

The use case starts when a user begins typing into the email and password text fields.

**Main flow**

1. The user inputs their email address <See A1, A2, A3, A4 & A5>.
2. The user inputs their password.
3. The system encrypts the user’s inputs with AES.
4. The system creates a connection to Firebase Authentication and Firebase Database. <See E1>
5. The system verifies that the email and encrypted passwords match. <See E2>
6. The system creates a session key and puts it into the cookies.
7. The user is brought to their news feed.

**Alternate flow**

A1 : User uses Facebook to login - Successful

1. The user clicks on “Facebook Login”
2. The user is taken to a Facebook Login Page <See A2>
3. The user enters their email and password for Facebook
4. The user is prompted by Facebook if they want to login to Social Modifications.
5. The user selects “Login”.

<Returns to number 6 in Main Flow>

A2 : User uses Facebook to login - Failed

1. The user clicks on “Facebook Login”
2. The user is taken to a Facebook Login Page <See A2>
3. The user enters their email and password for Facebook
4. The user has not entered the correct details.
5. The user clicks on Back

<Returns to number 1 in Main Flow>

A3 : User uses Facebook to login – User previously logged in with Facebook

1. The user clicks on “Facebook Login”
2. The user is prompted by Facebook if they want to login to Social Modifications.
3. The user selects “Login”.

<Returns to number 6 in Main Flow>

A4 : User registers - Database

1. The user clicks on “Register”
2. The user inputs their email address
3. The user inputs their password
4. The user inputs their password again <See E3>
5. The user enters in their name (or screen name)
6. User enters their date of birth <See E4>
7. The system creates a connection to Firebase Authentication and Firebase Database.
8. The system encrypts the password using AES.
9. The system inputs the details into the User table.

<Returns to number 6 in Main Flow>

A5 : User registers - Facebook

1. The user clicks on “Register”
2. The user clicks on “Facebook login”

<Returns to number 2 in A1>

**Exceptional flow**

E1 : System cannot connect to Firebase

1. The system cannot connect to Firebase Authentication and Firebase Database.
2. The system displays that technical difficulties are occurring.
3. The user is logged into a temporary account.
4. A unique temporary ID is displayed to the user

(Note: This is so the user can connect this temporary account to their original account. The system will merge data when this happens)

E2: Password not valid - Login

1. The system displays a prompt indicating the user’s email or password is not valid

<Returns to number 1 in Main Flow>

E3: Passwords do not match - Register

1. The system displays a prompt indicating the user’s passwords do not match.

<Returns to number 3 in A4>

E4: User not over 18 - Register

1. The system displays a prompt indicating the user is not over 18.
2. The user clicks “OK” on the prompt.
3. The system locks them out for 10 minutes.
4. The system displays message saying the user has been locked out.

<Returns to number 1 in Main Flow>

**Termination**

When the user is logged into SM, this use case ends.

**Post condition**

The system goes into a wait state

### Requirement 2: Add password

#### Description & Priority

This Use Case describes how a user can add a password if they registered an account using Facebook. This is an important feature as it will allow users to sign in if they lose access to their Facebook.

#### Use Case

**Unique ID:** addPass

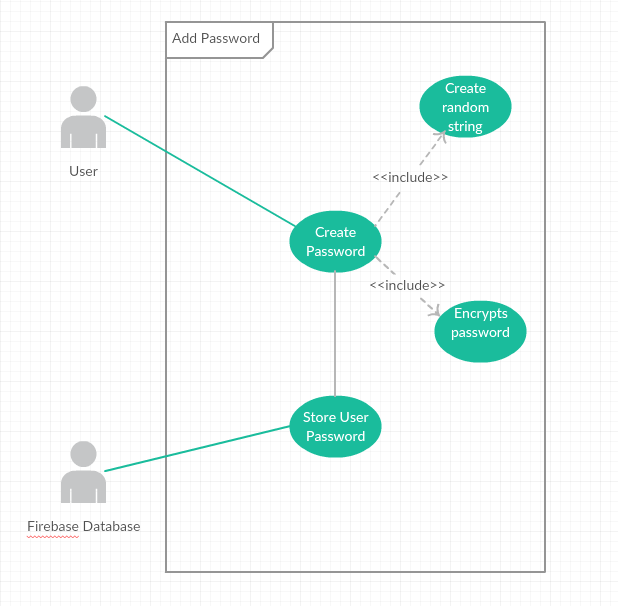
**Scope**

The scope of this use case is to allow a user to create a password if they created their account through Facebook.

**Description**

This use case describes the user to add a password to their account if they registered an account through Facebook. There is also a random string generator, so a user can use that as their password.

**Use Case Diagram**

****

**Flow Description**

**Precondition**

* The system is active and working correctly
* The user has created an account through Facebook
* The Firebase Database is connecting correctly.
* The Add Password page will be the next page the user sees after registering through Facebook.
* The user must create a password before continuing

**Activation**

This use case starts when a user creates an account through Facebook.

**Main flow**

1. The system identifies the ………….
2. The <Actor> …………...(See A1)
3. The system …………..(See E1)
4. The <Actor> ………….

**Alternate flow**

A1 : <title of A1>

1. The system …………..
2. The <Actor> ………….
3. The use case continues at position 3 of the main flow

**Exceptional flow**

E1 : <title of E1>

1. The system …………..
2. The <Actor> ………….
3. The use case continues at position 4 of the main flow

**Termination**

The system presents the next ……….

**Post condition**

The system goes into a wait state

**List further functional requirements here, using the same structure as for Requirements 1 & 2. Most systems would have at least five main functional requirements.**

## Non-Functional Requirements

Specifies any other particular non-functional attributes required by the system. Examples are provided below. **Remove the requirement headings that are not appropriate to your project.**

### Performance/Response time requirement

### Availability requirement

### Recover requirement

### Robustness requirement

### Security requirement

### Reliability requirement

### Maintainability requirement

### Portability requirement

### Extendibility requirement

### Reusability requirement

### Resource utilization requirement

# Interface requirements

This section describes how the software interfaces with other software products or users for input or output. Examples of such interfaces include APIs, web services, shared memory, data streams, and so forth. Most systems would have a GUI. Add more subsections for other interfaces as reuired.

## GUI

Include mock-ups of the key pages or stages of the system. Explain how they are linked. Explain how you addressed above requirements in the design. It is important that the mock-ups are in line with the functional requirements above, e.g., if one of your requirements is “user registration” then one of the screens listed in this section should show a registration page.

## Application Programming Interfaces (API)

Explain which interfaces your system offers or which are used by your system. Examples include Google maps and Weka.

# System Architecture

Use a class diagram to outline the structure of the system. Explain briefly why you have chosen this architecture. You might want to use Visio or Rational Rose to create these.

# System Evolution

This section describes how the system could evolve over time.