Software Requirements Specification

HamsterHelp

Version 2.1.3

Prepared by: Team 4

Duc Minh Le

Hung Phuc Chu

Jun Rong Brian Chong

Jun Shen Ban

Shivkumar Jaju

04/19/2018

Table of Contents

Revision History

Date	Changes	Version
02/05/2018	Initial Document	1.0
02/09/2018	Editing of the Document	2.0
02/23/2018	i. Removed Chat ii. Review and Rating pushed to Phase 5 iii. Features renamed (Eg: Framework System -> Framework)	2.1
03/16/2018	i. Tag Searching pushed to Phase 4 ii. Profile Searching pushed to Phase 4 iii. Removed “Tweaking” sections from Phase 4 iv. Register Page moved from Phase 5 to Phase 3	2.1.1
04/06/2018	i. Added Homepage in Phase 4 ii. Added Result pages in Phase 4	2.1.2
04/19/2018	i. Changed Review and Rating to Rating ii. SOS Call has been edited by changing from sending out email, mail and a phone call to emergency phone call.	2.1.3

1. Introduction

1.1. Purpose

The aim of this document is to explain the requirements, features and the details of HamsterHelp app.

1.2. Scope

HamsterHelp is a socializing and helping application for students(international and domestic students who might be coming from other country/state) who are new to the university and are seeking help in getting settled. The user will be able to gain new friends and get assistance to get settled in their new environment. The user will be bestowed a sorting feature that will allow the user to find a range of people based on the tags. The user will also have a rating system that helps other users to find a better helper. The user can also do profile searching which allows the user to specifically find somebody that they want. This app will be made using Android Studio.

1.3. Definitions, acronyms, and abbreviations

Android Nougat: A version of Android OS
Android Studio: An IDE made for making Android apps.
Google Play Store: An application that acts as a marketplace to download apps.
GUI: Graphical User Interface
IEEE: Institute of Electrical and Electronics Engineers.
ITIL: Information Technology Infrastructure Library.
MySQL: An open-source relational database management system.
OS: Operating System
RAM: Random Access Memory
Salt: A way of cryptography.
SOS: A type of Morse code for signaling danger.
ITIL: Information Technology Infrastructure Library

1.4. References

a) IEEE Recommended Practice for Software Requirements Specification, Template A.5, page 23, http://ieeexplore.ieee.org/document/720574/
b) Change Management, https://wiki.en.it-processmaps.com/index.php/Change_Management

1.5. Overview

The document will cover the overall description, specific requirements, and the appendix for the application. This document should describe the whole application to the reader with precision and clarity using the IEEE standard format. The overall description will cover the product perspective, product functions, user characteristics, constraints, assumptions and dependencies. Specific requirements will cover external interface requirements, functions, performance requirements, logical database requirements, design constraints and software system attributes. The appendix should cover the image of the look of the application.

2. Overall Description

2.1. Product perspective

2.1.1. System interface

There will not be any external system interfaces required to run in Android devices.

2.1.2. User interfaces

The user interface will be the touchscreen of Android devices for navigation in the application that will have a clean, simple, and attractive GUI.

2.1.3. Hardware interfaces

This application will be able to run on Android phone devices.

2.1.4. Software interfaces

The application can be downloaded from the Google Play Store for the latest version of Android Nougat, 7.0 or higher.

2.1.5. Communications interfaces

The application will be communicating with a MySQL database which contain the data.

2.1.6. Memory constraints

The application will be using a simple GUI and should take about 300MB of memory space. The application should take around 512MB of RAM.

2.1.7. Operations

The operations for this application will be running the application, interacting with a database and being able to search a data for the user.

2.1.8. Site adaptation requirements

There will not be any site adaptations in the application.

2.2. Product functions

The product function is to be a social application that allows the user to get help and make new friends.

2.3. User characteristics

This application will be a mixture of Facebook, WhatsApp and Tinder which are commonly used social applications. Social applications are apps that allow users to communicate with one another through the means of Internet. This application will require the user to login to personal accounts, similar to what most social application do. Not only that, this application enable users to obtain help and gain new friends at the same time. The application will be similar to these social application but tuned with a helpful approach. The user will be able to use this easily as most of the people have used social application before.

2.4. Constraints

The constraints of this application will be the screen resolution. Since most major smartphones has new screen technology like the Samsung Galaxy S8 and LG G6, we assume that most of the users do not have smartphones like these. So, the application can have a distorted version of the application but not the old ones. The application should not take much resources as this will potentially lag out the phone. Some smartphones will also have multitasking feature in them, potentially using the multitasking feature can break the application.

2.5. Assumptions and dependencies

Since we are building on an Android platform, as long as there are no major changes in the OS itself, the application will be able to run. We are also assuming that the MySQL database will not have a major change as well. The dependencies of this application will be the OS and the database.

2.6. Apportioning of Requirements

Due to our tight schedule, customer requirements might not be able to fulfill on schedule. So, to compromise, we will prioritize setting up the main functions required and the main features of the application. After that, during the agile iteration phase, new functions will be added if time permits. This can be referred in Section 3, Specific Requirements.

3. Specific Requirements

3.1. External interface requirements

This application will not have any external interface requirements.

3.2. Functions

3.2.1. Phase 1

3.2.1.1. Framework

3.2.1.1.1. The framework is the main code of the application that contains the necessary methods to make the application work as well as interacting with the database.

3.2.1.1.2. The input of this system will be the fields that the user entered in a system and the output of this will be the output of the specific system.

3.2.1.1.3. Additional Functional Requirements

3.2.1.1.3.1. The system will need to interact with the database (3.2.1.2) all the time.

3.2.1.1.3.2. The methods in the main code will need to interact with all the system in the application.

3.2.1.2. Database

3.2.1.2.1. The database is the main place where all the data for the application are stored, updated and cleared.

3.2.1.2.2. The input of this system will be the insertion of data while the output of it will be the movement of data which means getting, updating or deletion of data.

3.2.1.2.3. Additional Functional Requirements

3.2.1.2.3.1. The system will send a notification for every insertion, deletion, and updation of a user's data to the user.

3.2.1.2.3.2. The system will have logs that keep track of the flow of user's information for history keeping.

3.2.1.3. Login Page

3.2.1.3.1. The purpose of having this is to protect the user from anyone checking their information.

3.2.1.3.2. The input of this system will be the input of an email and a password. The output will be the access to the application.

3.2.1.3.3. Additional Functional Requirements

3.2.1.3.3.1. This system will need to interact with the database (3.2.1.2) for the verification of the existing user

3.2.1.3.3.2. This system will also need to interact with the register page (3.2.5.2.3)as well if the new user do not have a login access.

3.2.2. Phase 2

3.2.2.1. User Interface (GUI)

3.2.2.1.1. The user interface is an interface that allow users to easily understand the way to use the application and it provides an image for the application as well.

3.2.2.1.2. Additional Functional Requirements

3.2.2.1.2.1. The user interface will need to interact with the methods from the Framework.

3.2.2.1.2.2. The user interface will need to change when it goes from 1 page to the other.

3.2.3. Phase 3

3.2.3.1. Register Page

3.2.3.1.1. The purpose of this system is to allow the user to make an account for the access of the application.

3.2.3.1.2. The input of this system will be a name, age, phone number, email, password and current address. The output of this will be an account.

3.2.3.1.3. Additional Functional Requirement

3.2.3.1.3.1. The data from the fields of the new account will be stored in the database.

3.2.3.1.3.2. A verification email will be sent to the user's email to ensure safety and access of the application.

3.2.3.2. Tweaking of the Framework

3.2.3.2.1. The purpose of tweaking the framework system is to allow future methods to be crafted using the methods from the Framework system. This can be done using libraries and better code implementation.

3.2.4. Phase 4

3.2.4.1. Tags Searching

3.2.4.1.1. The purpose of tweaking the review and rating system is to allow better fluidity of the reviewing and rating process.

3.2.4.1.2. The input of this system will be the tags entered by the user. The output of this will be a list of other users that are found within the tags.

3.2.4.1.3. Additional Functional Requirements

3.2.4.1.3.1. The system will need to interact with the database (3.2.1.2) to get the data needed.

3.2.4.1.3.2. The system will also interact with the user interface (3.2.2.1) so that the data that is obtained from the database (3.2.1.2) is displayed in a list.

3.2.4.2. Searching Profile

3.2.4.2.1. The purpose of this system is to allow the user to specifically search a helper that they have in mind.

3.2.4.2.2. The input of this system will be an email while the output of this will be a specific helper's profile.

3.2.4.2.3. Additional Functional Requirement

3.2.4.2.3.1. The system will need to interact with the database system (3.2.1.2) for getting the data.

3.2.4.2.3.2. The system will also need to interact with the user interface (3.2.2.1) so that the data can be presented in a list.

3.2.4.3. Homepage

3.2.4.3.1. The purpose of this system is to allow the user to select the method of profile searching.

3.2.4.3.2. The input of this system will be selection between 2 buttons displayed to link to either Tag searching page or Email searching page.

3.2.4.4. Result pages

3.2.4.4.1. The purpose of this system is to receive inputs from user and display results from profile searches.

3.2.4.4.2. The input of this system will be selection on Tag searching page and email address on the Email searching page.

3.2.5. Phase 5

3.2.5.1. Tweaking of the GUI

3.2.5.1.1. The purpose of tweaking the GUI is to allow better user experience and to have a better looking application.

3.2.5.2. New Features

3.2.5.2.1. SOS Call

3.2.5.2.1.1. The purpose of this system is to allow the user to call the helper if the user is in any trouble.

3.2.5.2.1.2. The input of this will be a button implementation in the helper's profile. The output will be an immediate call to the helper.

3.2.5.2.1.3. Additional Functional Requirements

3.2.5.2.1.3.1. This will need to interact with the database system (3.2.1.2) to get the data of the helper.

3.2.5.2.1.3.2. The system will also be using the contact informations of the helper to send out emergency calls.

3.2.5.2.2. Rating

3.2.5.2.2.1. This system is to provide a feedback to the helper and to allow others to judge whether they want the helper or not.

3.2.5.2.2.2. The input of this system will be a rating using the scale from 1 to 5. The output of this will be the total average rating that will be posted on the helper's page.

3.2.5.2.2.3. Additional Functional Requirements

3.2.5.2.2.3.1. The system will need to interact with the database system (3.2.1.2) for data manipulation

3.2.5.2.2.3.2. The system will also need to interact with the user interface (3.2.2.1) so that the data can be displayed in a list.

3.3. Performance requirements

The application will be able to run without any crashes or lags in any Android smartphones.

3.4. Logical database requirements

The types of information that will be used by the functions are name, email, contact number, nationality, current address,and the place of origin. The functions has full accessing capabilities to these data. The frequency of use will be high for some methods but not all.

3.5. Design constraints

The main constraint of this system is that the application is made for Android platform. This means that Apple and Windows devices will not be able to access this application. Since we are building a mobile application, the space that we can use for a user interface is limited so our user interface needs to be space efficient to fully utilize the empty spaces for a purpose.

3.6. Software system attributes

3.6.1. Reliability

The application will be able to work without any problems during delivery

3.6.2. Availability

The availability of this application will only be in Android 7.0 or higher.

3.6.3. Security

The password of each user will be generated using Salt and store it into the database. Salt is a random data that is used to hash data, such as password or a passphrase. Each interaction of data in the application will be sent to the database as logs for history keeping.

3.6.4. Maintainability

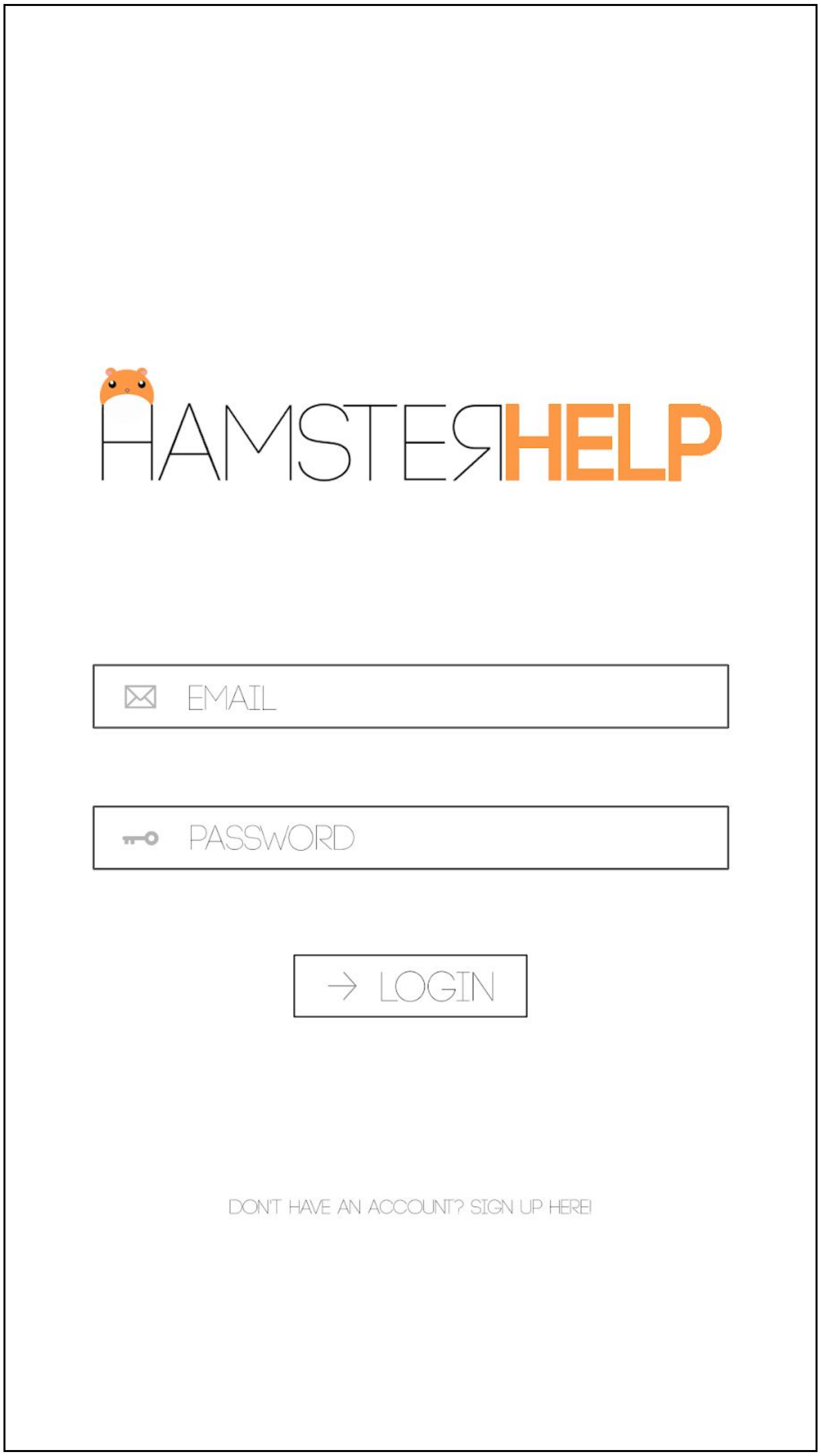
The system will be a redirection of a page to another so future features can be implemented easily as we will have a clean slate to work for each feature. Some of the features will be improved over time as well. The user interface will also improve over time for a better look and easy usage.

3.6.5. Portability

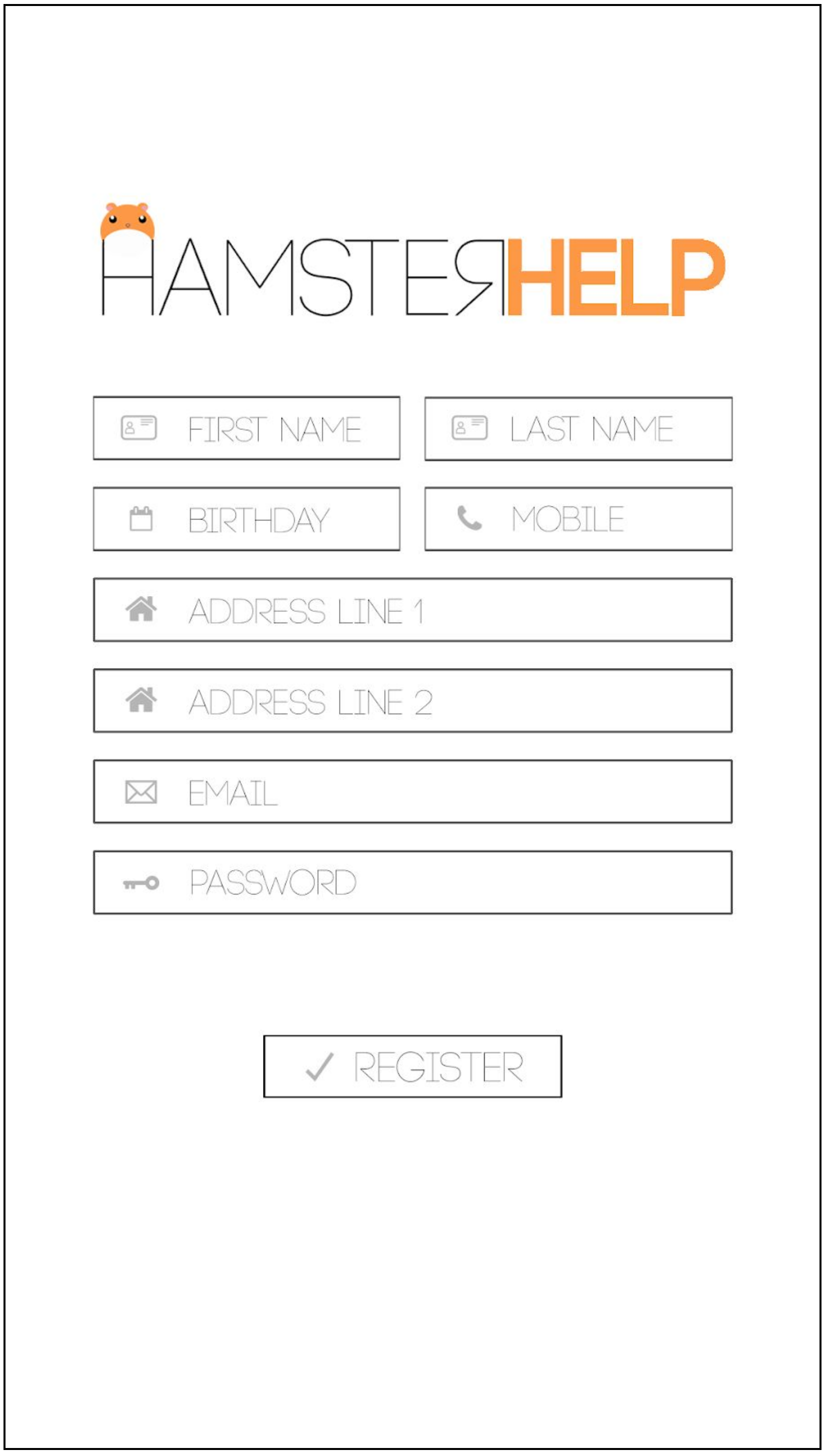
Since this application is made for Android, Android users can download the application through the Google Play Store using an Internet connection. The application will be able to run in Android OS that are version 7.0 and above.

4. Appendix

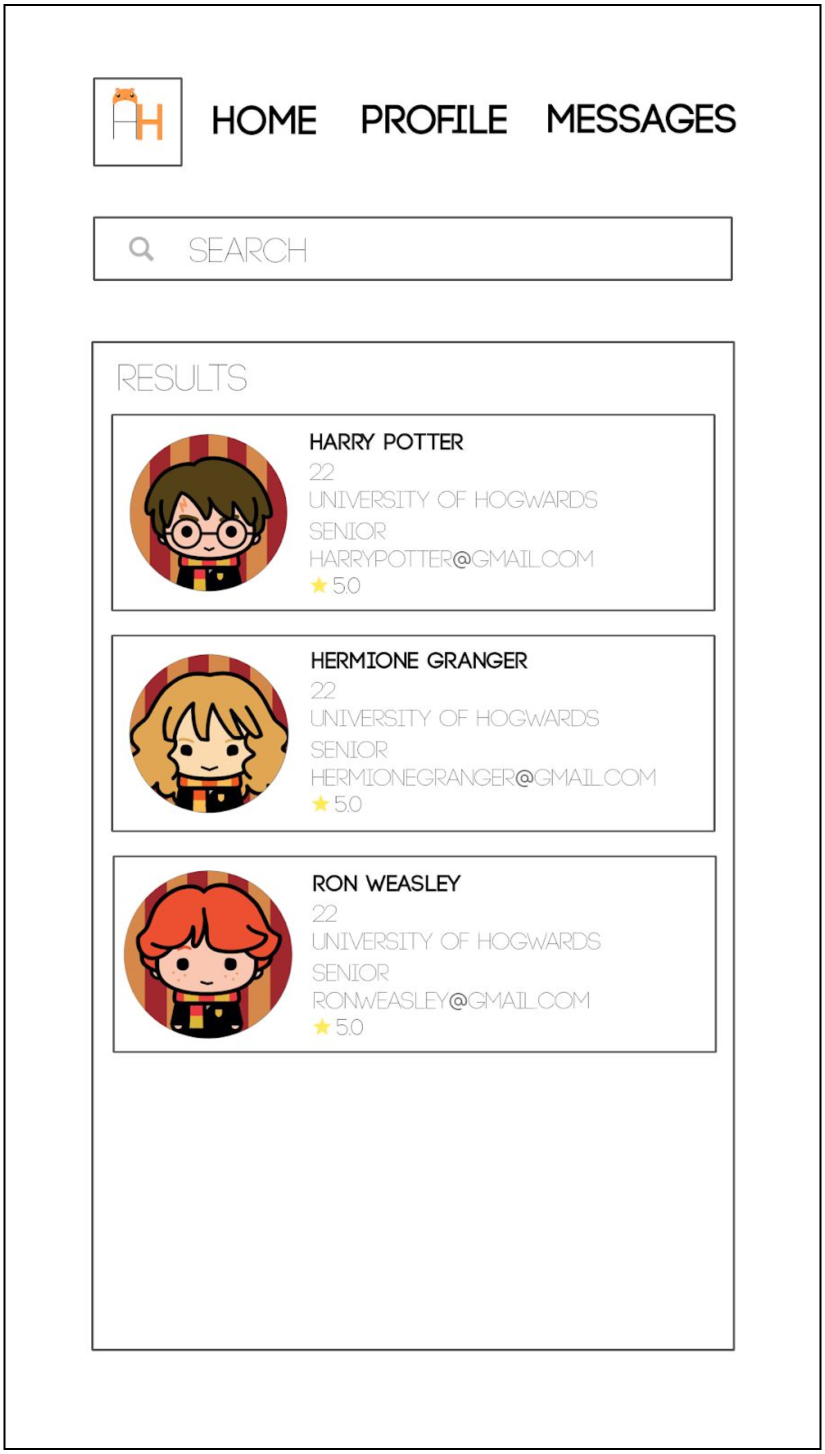
4.1 Login Page User Interface (GUI).



4.2 Registration Page User Interface (GUI).



4.3 Search Page User Interface (GUI).



RESULTS

HARRY POTTER
22
UNIVERSITY OF HOGWARDS
SENIOR
HARRYPOTTER@GMAIL.COM
★ 5.0

HERMIONE GRANGER
22
UNIVERSITY OF HOGWARDS
SENIOR
HERMIONEGRANGER@GMAIL.COM
★ 5.0

RON WEASLEY
22
UNIVERSITY OF HOGWARDS
SENIOR
RONWEASLEY@GMAIL.COM
★ 5.0

5. Change Management Process

If there are changes request to an implemented feature by customers, a representative from the software engineering team will get in contact with the customer to further discuss about the changes requested. To control the changes to the requirements, new features requested by the customer can be added into the existing features. Existing features can also be requested to be removed. The changes of the requirements can be sent via email or direct messages to the software engineering team. The change management process that will be used to identify, log, evaluate and update the SRS to reflect changes in the project scope and requirements will be the ITIL Change Management which is a change management process that enable beneficial changes to be made with minimum disruption to the SRS.

6. Document Approvals

____

Prof. Hamid Bagheri

Date: 3/16/18