CSCE 361

Honors Project

Colton Harper

Software Requirements Specification

Document

Version: (1.0) **Date: (10/06/2017)**

Table of Contents

1. Introduction

1.1 Purpose

This document is intended to assist those who design and implement the functionalities and software specifications that this system will require.

1.2 Scope

The software system that will be developed will consist of a front-end web interface and a back-end database. The web interface will be a public website that will allow local non-profit organizations and other community groups who are seeking software development help to enter their basic information; this information may include their contact information, a general description of who they are and what products or services they provide, and a brief description of what they are looking for or what software problems they may be facing. The goal of this system is to more easily connect local community groups who may have software needs to software engineering, computer engineering, or computer science students who are looking for projects to work on to develop their software engineering skills. The overarching aim of this system is to have a mutually beneficial impact on both the students developing the software tool and the local community groups by providing meaningful projects to students with real world stakeholders which will ideally hold the students more accountable. As a result, this added accountability may help them better develop their skills and deliver a better end-product to the community groups.

1.3 Overview

The remaining sections in this document describe the overall description and the specific requirements for this system. In particular, the overall description section will provide some information about the interfaces of the document, some functions and operations, some assumptions of the system, and the allocation of requirements based on implementation increment. In the specific requirements section, the specific requirements of the system are detailed out. These requirements include functional requirements, as well as, the performance, memory, and database requirements of the system.

2. The Overall Description

2.1 Product Perspective

2.1.1 System Interfaces

This system shall not be dependent upon any additional interfaces.

2.1.2 Interfaces

This community group information collection system will have a public web interface. Users may navigate to this web page which will allow the users to enter the requested information. There will also be an unlinked web interface that allow users to view the community group information that has been inputted into the system.

2.1.3 Hardware Interfaces

All personal computers that have the ability to run the Firefox, Chrome, and Safari web browsers should be able to navigate and use this system.

2.1.4 Software Interfaces

The system shall interface with the Firefox, Chrome, and Safari web browsers.

2.1.5 Communications Interfaces

There are no communication requirements for this system.

2.1.6 Memory Constraints

The meory usage of this web application will be minimal. There are no memory requirements for users of this system.

2.1.7 Operations

Users will navigate to the public website. Users may then view the information on the website about this software development initiative. Users will have the option to input their information via a web-form. Students can view the community groups' information via a "hidden" web page. In this case, hidden refers to a website that is not searchable via a search engine and is not hyperlinked to from another webpage. Special access users will have the ability to edit the status of the database entry, add notes, and remove entries.

2.1.8 Site Adaptation Requirements

This system will have no site adaptation requirments.

2.2 Product Functions

This system will allow community groups to enter some of their basic information.

This system will send a confirmation email to the community group once their information has successfully been entered.

This system will allow students wishing to develop a software system for a community group to view the listings that have already been entered into the system.

2.3 User Characteristics

There are two types of users that will be interacting with the system: community groups and students. Community groups will use the system to inform students that they are seeking software development assistance and to provide the students with their contact information. Students will use the system to retrieve information about community groups seeking software development assistance.

2.4 Constraints

This system will be designed for personal computers and will not support a mobile-friendly format for handheld devices.

2.5 Assumptions and Dependencies

This system relies on its compatibility with Firefox, Chrome, and Safari web browsers. If there are no substantial changes to these web browsers the system should work and display properly.

The ability for students to view community group information is dependent upon community groups consistently populating the database by entering their information in the web form.

2.6 Apportioning of Requirements.

In the first increment of this project a database will be developed to store community group information. A basic web interface will be created to allow community groups to enter the requested information. The website will have information regarding this software development initiative. The web interface will also have general webform functionality, which will allow users to input the information to the database. In this increment there will also be a page created where students may view the community groups' information.

In the future increments of this project, the aesthetics of the user interface will be augmented. Furthermore, functionality for a special class of users will have the ability to mark the status of, write notes about, and remove entries in the database.

3. Specific Requirements

3.1 External Interfaces

This system will not use any external interfaces.

3.2 Functions

3.2.1. Phase I

3.2.1.1. Main Web Page

3.2.1.1.1. Description: The web application shall have a main page that contains the initiative description.

3.2.1.1.2. Description: The web application shall have a main page that contains a web form that records community group information.

3.2.1.2. Web Form

3.2.1.2.1. Description: The web form shall allow users to input basic community group information including community group contact information, a general description of who they are, what products or services they provide, and what assistance they are looking for. Alternatively, the community group may leave a description of what software problems they may be facing.

3.2.1.2.2. System Response: Upon submitting group information, the system shall add the information to the database.

3.2.1.3. Responsiveness to desktop

3.2.1.3.1. Description: The web application shall be formatted for desktop displays.

3.2.1.4. Email Confirmation

3.2.1.4.1. Description: The system shall send a confirmation of submission email to the user.

3.2.1.4.2. System Response: Upon successful submission of community group information the user will be sent an email that confirms their entry was successfully submitted.

3.2.1.4.3. Dependencies: 3.2.1.2

3.2.1.5. Group Information Page

3.2.1.5.1. Description: The system shall contain a web page that displays the community group information.

3.2.1.5.2. Description: The system shall have the functionality to search for community groups in the data set.

3.2.1.5.3. Description: The system shall have functionality to sort the displayed information.

3.2.1.6. Searching Group Information

3.2.1.6.1. Description: The system shall have a searching function that allows users to search for community groups by name.

3.2.1.6.2. System Response: Upon entry of a search string into the search box, the system shall return results that match the search string.

3.2.1.6.3. Dependencies: 3.2.1.5

3.2.1.7. Sorting Group Information

3.2.1.7.1. Description: The system shall have a sorting function that allows users to choose to sort the community group information by name or by date.

3.2.1.7.2. System Response: Upon selecting sort type, the system shall sort the data query results and display the corresponding sorted entries.

3.2.1.7.3. Dependencies: 3.2.1.5

3.2.1.8. Mock Data

3.2.1.8.1. Description: The system shall be set up so that mock data can populate the system for testing purposes.

3.2.2. Phase II

3.2.2.1. Special Access Login

3.2.2.1.1. Description: The system shall provide a method for special access users to create accounts to perform restricted functions.

3.2.2.1.2. Description: The system shall provide a method for existing users with special access to login to their accounts.

3.2.2.1.3. System Response: The user may select the login option. The system shall redirect the user to the login page.

3.2.2.1.4. System Response: The user may enter their credentials, or create a new account and then enter their credentials. The system shall then redirect the user to the restricted access page if the correct credentials are valid.

Restricted Access Page

3.2.2.1.5. Description: The system shall have a web page for users with special access.

3.2.2.2. Restricted Access Functions

3.2.2.2.1. Description: The system shall support additional functions to edit the status of, add notes to, and remove existing entries in the database.

3.2.2.2.2. System Response: Upon accessing the restricted access page, selecting a community group information entry, and choosing one of the special access functions (edit status, add notes, remove), the system shall update the database accordingly.

3.3 Performance Requirements

The web form entries will update the database and will be viewable by students upon 20 seconds from the time of the entry submission. The system shall accommodate for up to 50 submissions at once. In normal conditions, the workload will be zero to one submissions per minute. In heavy workload conditions, the system may take 50 submissions per minute.

3.4 Logical Database Requirements

The database used in this system will be updated every time a user submits information in the form. The database will also be accessed when students retrieve information to view the community groups' information. The database will keep information about the community groups until the community group information is removed by users with special access. The data stored in the table of this database will be of string and numerical types.

3.5 Design Constraints

This system will be designed for desktop browsers and will therefore not have a mobile-friendly format.

3.6 Software System Attributes

3.6.1 Availability

The website shall be available to access any time of the day.

3.6.2 Security

The information that is being stored in this database should not be sensitive data. The website will clearly inform the users not to include sensitive information. Although the page is “hidden” it can still be publicly accessed if a user obtains the corresponding link.

In future phases, the system will have functionality to update the status of, write notes about, and remove entries in the database. These functionalities will have restricted access. Therefore, a login system will use strong hashing algorithms like SHA-1 to encrypt passwords. The database will be protected from malicious SQL injections.

3.6.3 Maintainability

Future additions to this system, such as adding functionality to remove entries, shall not need much refactoring of the existing system and can be simply added on top of the existing system.

3.6.4 Portability

Porting the database should be a simple process with the following steps in place. A portable file type that preserves the SQL database structure will be backed up. The website would need to then submit and retrieve data from the new location.