# Software Requirements

# Software Requirements Specification (SRS)

Revision History:

|  |  |  |
| --- | --- | --- |
| Date | Author | Description |
| 3-20 | Yan Jiaxi | Adding/Editing Use Cases |
| 3-28 | Yan Jiaxi | Adding Use Cases to the Disabled and the Elderly |
| 3-29 | Yan Jiaxi | Adding Use Cases and System Abilities of Backstage Management for Administrators |
| 4-14 | Rui Pinto and Tiago Morais | Add Use Cases and change names of Use-Cases to match with the diagram |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Contents

[Software Requirements 1](#_Toc132381812)

[Software Requirements Specification (SRS) 1](#_Toc132381813)

[1.  Introduction 3](#_Toc132381814)

[1.1    Intended Audience and Purpose 3](#_Toc132381815)

[1.2    How to use the document 3](#_Toc132381816)

[2.  Concept of Operations 4](#_Toc132381817)

[2.1    System Context 4](#_Toc132381818)

[2.2 System capabilities 4](#_Toc132381819)

[3.  Use Cases 5](#_Toc132381820)

[Case 1: User Wants to disconnect the equipment 5](#_Toc132381821)

[Case 2: User Wants to Register a New Account 5](#_Toc132381822)

[Case 3: User Wants to Log in to the Web Station 6](#_Toc132381823)

[Case 4: User Wants to View User Information 6](#_Toc132381824)

[Case 5: User Wants Filter Data 6](#_Toc132381825)

[Case 6: User Wants to View Usage Guide 7](#_Toc132381826)

[Case 7: User Wants to Know the information of the algorithm 7](#_Toc132381827)

[Case 8: User Wants to Know the advantage of the model 7](#_Toc132381828)

[Case 9: Administrators Wants to View the List of Devices 8](#_Toc132381829)

[Case 10: Administrators Wants to View the Historical Data 8](#_Toc132381830)

[Case 11: Administrators Wants to Manage Users’ Information 8](#_Toc132381831)

[Case 12: Administrators Wants to Put a Notice on the Web Site 9](#_Toc132381832)

[Case 13: Administrators can Manage the System Log 9](#_Toc132381833)

[Case 14: View Real-time Information 9](#_Toc132381834)

[4.    Behavioral Requirements 10](#_Toc132381835)

[4.1 System Inputs and Outputs 10](#_Toc132381836)

[4.2 Detailed Output Behavior 10](#_Toc132381837)

[4.3   Quality Requirements 11](#_Toc132381838)

[5.    Expected Subsets 11](#_Toc132381839)

[6.    Fundamental Assumptions 11](#_Toc132381840)

## 1.  Introduction

### 1.1    Intended Audience and Purpose

This document is intended to provided information development process, ensuring that all system requirements are met. The following entities may find the document useful:

Primary Customer - This page will detail all of the web page requirements as understood by the production team. The customer should be able to determine that their requirements will be correctly reflected in the final product through the information found on this page.

User - A prospective user will be able to use this document to identify the main functionality included in the web page. Furthermore, the web page will have a set of system requirements before the web page can be run. Details regarding these requirements can be found here.

Development Team - Details of specific requirements that the final software build must include will be located here. Developers can use this document to ensure the software addresses each of these requirements.

QA Team - By developing testing procedures founded in the system requirements, the QA Team can create a comprehensive testing regimen that will guarantee requirements are met.

### 1.2    How to use the document

Table of Contents:

1. Introduction

2. Concept of Operations - broad description of the purpose of the web pages

  2.1 System Context - details any specific system requirements the web pages will require to run

  2.2 System Capabilities - description in prose of all capabilities available to the user in the interaction

  2.3 Use cases - A detailed look at each functional requirement, describing the web pages context both before and after an action is taken

3. Behavioral Requirements - How the web page will interact with a user

  3.1 Input and output requirements - A description of allowed inputs and generated outputs

    3.1.1 Input - Describes any restrictions that will be placed on allowed input

    3.1.2 Output - Describes the range of outputs that can be generated

  3.2 Detailed Output Behavior - Output descriptions in prose

4. Quality Requirements - Requirements not pertaining to the function of the web page will be listed here

5. Expected Subsets - Expected levels of functionality at checkpoints during development

6. Fundamental Assumptions - Some specifics about input, output, or behavior upon which other requirements are founded will be listed here

7. Expected Changes - Future features and directions the project is expected to take

## 2.  Concept of Operations

  The goal is to create a user friendly assistive interaction of the sensor. It will allow its users to manage prosthetics and help users use them better. The web page uses document box, the web page page, and the jump between web page pages implemented by buttons to interact with the user. For more details on the usage and capabilities of the web page read the section, [System Capabilities](https://uocis.assembla.com/spaces/cis422w18-team2/wiki/Software_Requirements#System%20Capabilities).

### 2.1    System Context

**System Requirements(Not Functional):**

Requires a system with Web pages display because all of the operations are performed through these pages.

**Supported Browsers:**

* Chrome
* Firefox
* Opera
* Brave

**Supported Language:**

* HTML
* CSS
* JavaScript
* Ajax Bootstrap

### 2.2 System capabilities

This auxiliary interaction is a standalone program. When the user uses it, they need to log in to enjoy the services provided by the auxiliary interaction. If the user does not have an account, they need to register an account by jumping to the registration page and when they complete registration then they can use this auxiliary interaction.

After the user logs in, they enter the page which the user can see all the charts which reflect the predictions, data Storage in Database or other stuff.

After the users log in, they can see the introduction of the algorithm and the display of the model. They can understand the advantages of this algorithm and the security of our research and development products.

Users can view and modify personal information by going to the personal information interface through the personal information button in the menu. Users also can view the help Information window which displays guidelines for using the interactive system.

The background administrator can access the management page by logging into the administrator account. After they logging into the administrator’s account, they can search for the system monitoring to view the list of all the device and see the historical data diagram the do the delete operation. And who can login the administrator’s account can also manage the system to add, delete, revise, import or derive the users’ account. They can manage notification announcements and logs also.

## 3.  Use Cases

### Case 1: User Wants to disconnect the equipment

**Actors:**Administrator

**Goals:**The end user wants to close the pages and any opened equipment.

**Preconditions:**The web page is open and running.

**Case:**

1.1 From the File menu, the end user selects the "Close all and Quit" option.

1.2 The server disconnects from all currently added equipment

1.3 The pages terminates itself.

**Alternate Flows:**

1.2.1 Collected data have not uploaded or discarded

         The user is warned about quitting before uploading or discarding the data

      1.2.1.1 The user decides to upload data.

                  The web page closes after all data is uploaded.

  1.2.1.2 The user decides to discard data.

                  Discard data. The web page terminates.

**Exception Flows:**

1.2.2 The user forces the termination (by shutting down their machine, using Task Manager to Force Quit, etc)

         All data that is not uploaded or discarded is discarded.

**Postconditions:**If a user navigates to Task Manager (or equivalent process manager), there is no trace of the pages still run on Browser. No data should be displayed the next time the user opens web page.

### Case 2: User Wants to Register a New Account

**Actors:** Administrator

**Goal:** The user would like to be applied for a new account by entering personal information.

**Preconditions:** The web page is opened.

**Case**:

2.1 From the login interface, the end user selects the "Register" option.

2.2 The window turns to the register page which includes some personal information (e.g. email, password, etc.) input boxes.

2.3 Users can enter personal information.

2.4 User submit personal information.

2.5 The page reflects the result wrong or success.

**Alternate Flows**:

2.3 User chooses to return to the login page.

2.4.1 If the user is successfully registered, the login interface is returned.

2.4.2 If the user fails to register, let the user improve the information.

**Exception Flows:**

2.3.1 The user has not entered all the personal information or some of the information is not in the correct format or is not valid.

         The user is warned about submit before filling in all personal information correctly.

2.3.2 The account name already exists

         The user is prompted to change account name.

**Postconditions:**User gets a new account and can use it to log in to the web station.

### Case 3: User Wants to Log in to the Web Station

**Actors:** Administrator

**Goal:**The end user would like to log in to the web station so that they can add equipment and collect data.

**Preconditions:** The log-in web page is opened.

**Case:**

3.1 From the login interface, the end user enters account number and password.

3.2 User selects the "Login" option.

**Alternate Flows:**

**Exception Flows:**

3.2.1 If the account or password is incorrect or invalid. The end user will be warned by the page.

         The end user is prompted to change account or password.

**Postconditions:** The web page is running, waiting for its next instruction from the end user.

### Case 4: User Wants to View User Information

**Actors:**Administrator

**Goals:**The end user would like to view and change the user information.

**Preconditions:**The page is opened.

**Case**:

4.1 The end user enters the "User Information" interface

4.2 A new window opens with all user information

4.3 The end user can view all information.

**Alternate Flows:**

4.3.1 The end user can select and change one or more user information, such as passwords, etc.

         The end user confirms their changes. The web page saves all changed information

**Exception Flows:**

4.3.2 The information after user changes is invalid or incorrectly formatted. The end user will be warned by the page.

         The end user is prompted to change the information.

**Postconditions:**User information should be permanently changed and the next time the end user views user information or logs in, the changed information should be displayed.

### Case 5: User Wants Filter Data

**Actors:**Administrator

**Goals:**The end user would search with the name of the chart that he wants to see it.

**Preconditions:**The web page is opened on the user's browser.

**Case:**

5.1 The end user writes the name of the chart in search box

**Postconditions:**The page is open, waiting for its next instruction from the end user.

### Case 6: User Wants to View Usage Guide

**Actors:** Administrator

**Goals:** **The user is able to access information about the function of each function key**.

**Preconditions:**The web page is open and running.

**Case**:

6.1 The user hovers his mouse over a function key.

6.2 web page shows function information of the function key

6.3 User views the information and gain help.

**Alternate Flows:**

**Exception Flows:**

**Postconditions: User should be able to obtain detailed function instructions when user’s mouse hovers over a function key**

### Case 7: User Wants to Know the information of the algorithm

**Actors:** Administrator

**Goals:** **The user is able to access information about the advantages and effects of the algorithm.**

**Preconditions:**The web page is open and running.

**Case**:

7.1 The user hovers his mouse over a algorithm key.

7.2 web page shows algorithm information of the **advantages of the algorithm.**

7.3 User views the information.

**Alternate Flows:**

**Exception Flows:**

**Postconditions: User should be able to obtain detailed information about the algorithm when user’s mouse hovers over a algorithm key**

### Case 8: User Wants to Know the advantage of the model

**Actors:** Administrator

**Goals:** **The user is able to access information about the advantages of the model compared with other models.**

**Preconditions:**The web page is open and running.

**Case**:

8.1 The user hovers his mouse over a introduction key.

8.2 web page turns to the introduction page and shows the information of **the advantages of the model compared with other models.**

8.3 User views the information.

**Alternate Flows:**

**Exception Flows:**

**Postconditions: User should be able to obtain detailed information about the advantages of the model compared with other models. when user’s mouse hovers over a algorithm key**

### Case 9: Administrators Wants to View the List of Devices

**Actors:** Administrator

**Goals:** **The** Administrator **is able to access information about the devices which are connecting with the system .**

**Preconditions:**The management web page is open and running.

**Case**:

9.1 The user hovers his mouse over the devices key.

9.2 Web page shows devices linking with the system**.**

9.3 Administrators views the information.

**Alternate Flows:**

**Exception Flows:**

**Postconditions: administrators should be able to obtain detailed information about the devices.**

### Case 10: Administrators Wants to View the Historical Data

**Actors:** Administrators

**Goal:** **The** Administrator **is able to access information about the historical data.**

**Preconditions:** The management web page is opened.

**Case**:

10.1 The administrator hovers his mouse over the history key.

10.2 Web page shows the system historical data**.**

10.3 Administrators views the data.

**Alternate Flows**:

10.4 Administrator chooses to delete the history data.

10.5 If the administrator is successfully deleted, the page is returned.

10.6 If the administrator fails to delete, system prompt for deletion failed.

**Exception Flows:**

**Postconditions:**administrators can view the historical information and delete them.

### Case 11: Administrators Wants to Manage Users’ Information

**Actors:** Administrators

**Goal:** **The** Administrator **is able to manage information about the users.**

**Preconditions:** The management web page is opened.

**Case**:

11.1 The administrator hovers his mouse over the users management key.

11.2 Web page shows the users’ information and the add, delete, revise, import and derive keys**.**

11.3 Administrators views the data and choose the keys.

**Alternate Flows**:

11.4.1 Administrator chooses to add the users data.

11.4.2 Administrator fill the information about users and add it to the database.

11.5.1 Administrator chooses to delete the users data.

11.5.2 Administrator choose the information about users he want to delete and remove it from the database.

11.6.1 Administrator chooses to revise the users data.

11.6.2 Administrator choose the information about users he want to revise,revise it and renew it to the database.

**Exception Flows:**

**Postconditions:**administrators can manage the users’ information in the system.

### Case 12: Administrators Wants to Put a Notice on the Web Site

**Actors:** Administrators

**Goal:** **The** Administrator **is able to** put a notice on the web site.

**Preconditions:** The management web page is opened.

**Case**:

12.1 The administrator hovers his mouse over the notice adding key.

12.2 Web page turns to the page which has the function to edit a new notice**.**

12.3 Administrators add the new notice on the website.

**Alternate Flows**:

**Exception Flows:**

**Postconditions:**administrators can add a new notice on the web.

### Case 13: Administrators can Manage the System Log

**Actors:** Administrators

**Goal:** **The** Administrator **is able to** system log.

**Preconditions:** The management web page is opened.

**Case**:

13.1 The administrator hovers his mouse over the log management key.

13.2 Web page turns to the page which shows the list of the logs**.**

13.3 Administrators click the set key.

13.4 Administrators can revise or delete the log.

13.5 Administrators click the reserve button to reserve the log.

**Alternate Flows**:

**Exception Flows:**

**Postconditions:**administrators can manage the system logs.

### Case 14: View Real-time Information

**Actors:** Administrator

**Goal:** **The user can see real time updated data.**

**Preconditions:** The web page is opened.

**Case**:

14.1 The user clicks in the button to see charts.

14.2 Web page turns to the page which shows the charts**.**

14.3 The system updates the information every 3 seconds.

**Alternate Flows**:

**Exception Flows:**

**Postconditions:** administrators can constantly watch the data.

## 4.    Behavioral Requirements

### 4.1 System Inputs and Outputs

#### 4.1.1 Inputs

Inputs to the web page come from the user. Users can select options such as personal information, device information, help, and so on from the main menu. On the login screen, you can enter the text box, and on the personal information screen, you can choose to modify the information.

Input

Inputs When Editing Personal information:

\*account number: An account number that is unique to each user, consisting of letters and numbers

\*password: A string consisting of at least n digits number and characters to whether the guest is the user

\*user name: A contact can have a first name, or last name, or both. The first and last names will be separate fields. The system will accept any characters in the first and last name fields.

\*user information: The information provided by the user, which is mainly the user's contact information and physical condition

#### 4.1.2 Outputs

The Web pages display all information, such as user information, device information, help information and each windows contains any created contacts and interactable buttons for the user.

Outputs to The User:

\*data record: The program starts to collect data, the "Stop Collecting" and "Re-Collecting" buttons appear new, if you re-collect or stop collecting data, the system will generate a window asking whether to discard the data.

\*personal information of user: user detailed information, including the account number, email, real name, body information and so on. Users can click the "Modify Information" button to modify the information and display it.

\*help information: the window contains the instruction manual to guide the user to use the system

### 4.2 Detailed Output Behavior

The output is a GUI that provides the user with the ability to give inputs to the interaction.

After the user logs in,we can see any charts or data show in the page if we have collected the data on the mobile terminal.

Through the "User Information" and "Help Information" buttons on the main menu, users can view the user information page and the help information page. The user information page displays the user account, real name, physical condition and other related information, and the bottom of the page is composed of a "Modify Information" button, where users can modify personal information. The Help Information screen displays guidelines for using the interactive system.

### 4.3   Quality Requirements

The web page must be competitive with similar web pages in regards to performance, reliability, consistency, and scalability.

Performance: Responsiveness to user input

      \* Standard actions that manipulate interaction should not exceed 500ms execution time.

Reliability: Confidence that actions taken will not result in errors, and that changes made to program are persistent

      \* User input should not produce faults or errors that impact or hinder use of the web page

      \* Any modifications to program should produce a lasting change that persists through any following series of actions taken by the user.

Consistency: Persistent data in program contents

      \* program contents should be modifiable after being added in such a way that target fields can be changed without affecting data in other fields.

Scalability: Ease of extending web page capabilities

      \* web page should be modularized such that adding/extending features and functions only require changes to a single component and the interface with that component, if applicable.

## 5.    Expected Subsets

L0:

-Basic GUI with ability to check the device and view device details

-Ability to register and sign in

-Ability to add new device and delete useless device

-Ability to modify user information

-Ability to control data

L1:

-Complete GUI for making the interface user-friendly

-Complete the Administrator module

## 6.    Fundamental Assumptions

The web page can run on any system that is capable of running Qt.

The web page will not terminate when all windows are closed.

Software updates will be downloaded by the end user as opposed to pushed out by the developers.