Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering 2**

BeHealthy

Software Requirements Specifications

Donia Khaled

Safwa Salman

Mahmoud Samy Saeed

Hager Sobeah

Fall-2016

Contents

[Team 3](#_Toc447313429)

[Document Purpose and Audience 3](#_Toc447313430)

[Introduction 3](#_Toc447313431)

[Software Purpose 3](#_Toc447313432)

[Software Scope 3](#_Toc447313433)

[Definitions, acronyms, and abbreviations 3](#_Toc447313434)

[Requirements 4](#_Toc447313435)

[Functional Requirements 4](#_Toc447313436)

[Non Functional Requirements 4](#_Toc447313437)

[System Models 4](#_Toc447313438)

[Use Case Model 4](#_Toc447313439)

[Use Case Tables 5](#_Toc447313440)

[Ownership Report 6](#_Toc447313441)

# 

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20146011 | Mahmoud Samy | mahmoudsamy995@gmail.com | 01097690138 |
| 20146013 | Hager Sobeah | hagersobeah@gmail.com | 01202590820 |
| 20146018 | Donia Khaled | doniaakhaled@gmail.com | 01001366636 |
| 20146009 | Safwa Mohammed |  |  |

# Document Purpose and Audience

* **Software Requirements Specification is a Formal document that describes the functionality of the software using the use cases model and table.**
* **Expected Audience : Product Owner - Customer**

# Introduction

# Software Purpose

**The Software Purpose is to provide the User with remote control on all home appliances through an application that allow to turn on / off any device on the board and view the status of each appliance to decide whether it works probably or it needs to be modified.**

# Software Scope

* **Control home appliances through sensors embedded in it**
* **Add Member or Device to the board**
* **Switch on/off appliances on the board**
* **Remove Member or Device from the board**
* **View Current Status of home appliances**
* **Cut the electic current from home circuits**
* **Automatically Switch on Fire extinguisher in case of fire**
* **SHCS Software will be available on Android Devices and as Web App**

# Definitions, acronyms, and abbreviations

* **None**

# Requirements

# Functional Requirements

* **The Admin user first create an account to login and add users and devices.**
* **Open / close a device through the board.**
* **Add / remove a new device to the board.**
* **Add / remove a new member to the board.**
* **Enable & Disable the electric current flowing in home appliances.**
* **Give Alarms or warning notifications in emergency cases such as fire or failure in some devices.**
* **View the board status and the status of each device.**
* **The user searches for compatible devices in the house and adds them to the Smart Board.**
* **The home page will show the status of each machine at the current time. By clicking on the name of each device another page opens up with all options that could be done at the current state.**
* **The device connection could be suspended without having to remove the device.**

# Non Functional Requirements

**1 – Scalibilty :**

* **response to many requests at a time.**
* **System handles smoothly up to 20 devices per board.**

**2 – Documentation : to be as a guide for users.**

**3 - Compatibility : Compatible with android and web platforms.**

**5 - Response Time : at most 5 seconds for each request.**

**6- Disaster Recovery : response to emergency cases without having the permission of the Adminstrator.**

**7 - Configuration Management : managed by administrator**

**8 - Security : providing Security for the board to not to be penetrated.**

**9 - Reliability : software's ability to function, given environmental conditions, for a particular amount of time**

**10 -Usibility :startup a use guide will load to go through the whole system and how it could be used**

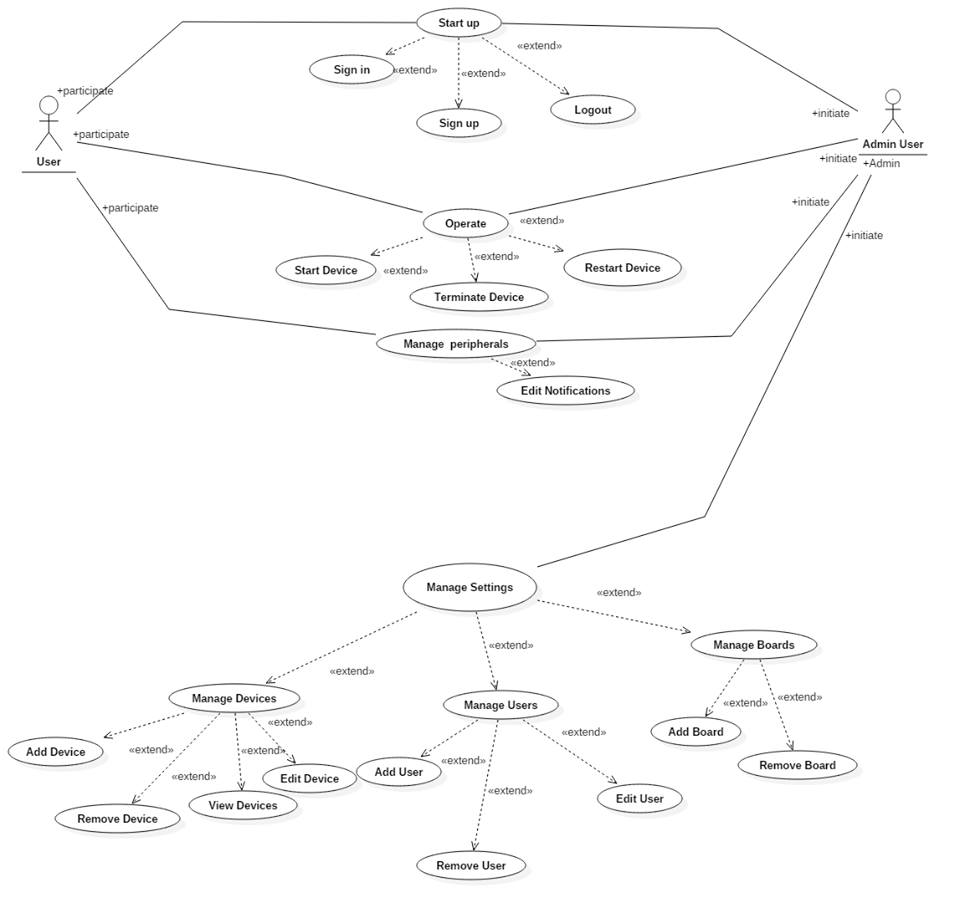
**11 - Performance**

* **Turning On / Off device operation is done in atmost 5 seconds.**
* **Update the Status of the board is done in atmost 3 seconds.**
* **Response to emergency cases in atmost 1 second.**
* **Adding or removing Member / device Operation is done in atmost 3 seconds.**
* **Cutting electric current operation is done in atmost 1 second.**

# System Models

## Use Case Model

## 



## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Sign in | |
| Actors: | User | |
| Pre-conditions: | User starts App | |
| Post-conditions: | User Sign in the app successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter Username and Password. |  |
|  | 2- System checks database |
| 3- User sign in successfully |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters username and password. |  |
|  | 2- e-mail or password is incorrect.  3- System rejects entry.  4-Asks user to enter data again. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Sign up | |
| Actors: | User | |
| Pre-conditions: | User starts App | |
| Post-conditions: | User Signs up successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enters e-mail, password and other details. |  |
|  | 2- System verifies data. |
| 3- User signs up successfully |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters e-mail, password and other details. |  |
|  | 2- System checks database.  3- System rejects entry.  4-Asks user to enter data again. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Sign Out | |
| Actors: | User | |
| Pre-conditions: | User is signed in | |
| Post-conditions: | User logout successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1- User presses sign out button. |  |
|  | 2- System allows the user to sign out and saves a backup of his data. |
| Exceptions: | **User Action** | **System Action** |
| 1- User presses logout button. |  |
|  | 2- System fails to save a backup. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Add Device | |
| Actors: | Admin User | |
| Pre-conditions: | Admin User press manage devices button. | |
| Post-conditions: | Admin User add a device successfully. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Press Add button. |  |
|  | 2- System starts searching for the device. |
| 3- Admin User confirm that the addition of the device. |  |
|  |  | 4- SHCS connect the device onto the board |
| Exceptions: | **User Action** | **System Action** |
| 1- User chooses a device. |  |
|  | 2- System fails to connect the device. |
|  | 1- User search for an unsuitable device. |  |
|  |  | 2- System tries to connect the device but every time it fails. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Checkup Reminder | |
| Actors: | User | |
| Pre-conditions: | A checkup time is already assigned. | |
| Post-conditions: | A checkup notification is sent to the user. | |
| Flow of events: | **User Action** | **System Action** |
|  | System sends the checkup notification at the exact time that was already assigned |
| Exceptions: | **User Action** | **System Action** |
|  | 1-System sends a checkup notification |
| 2-no response from the user |  |
|  |  |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Edit Device | |
| Actors: | Admin User | |
| Pre-conditions: | Admin User press manage devices button. | |
| Post-conditions: | Admin User edit a device successfully. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Admin User Press Edit button. |  |
| 2- Admin User chooses a device to edit. |  |
| 3- Admin User edits the device. |  |
|  |  | 4- SHCS update the device details. |
| Exceptions: | **User Action** | **System Action** |
| 1- Admin User chooses a device. |  |
| 2- Admin User make an unsuitable change in details. |  |
|  |  | 3- System fails to update device details |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | View devices | |
| Actors: | Admin User / User | |
| Pre-conditions: | Admin User / User press view devices button. | |
| Post-conditions: | Admin User / Userviews all devices connected to SHCS on current board. | |
| Flow of e-vents: | **User Action** | **System Action** |
| 1- User Press View devices button. |  |
|  | 2- System shows up all devices connected at the current time |
| Exceptions: | **User Action** | **System Action** |
| NULL |  |
| Includes: | NULL | |
| Notes and Issues: | NULL | |
| Use Case ID: | 8 | |
| Use Case Name: | Add pill | |
| Actors: | User | |
| Pre-conditions: | User enter pill name and time. | |
| Post-conditions: | Pill name and time is added. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enter the pill name and time. |  |
|  | 2- System checks if the pill name is in the database. |
|  | 3-the pill name and time is selected. |
| Exceptions: | **User Action** | **System Action** |
|  | 1-pill name isn’t in the database. |
|  | 2- System adds the pill name and time to the database. |
|  |  | 3-the pill name and time is selected. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Terminate Device | |
| Actors: | Admin User / User | |
| Pre-conditions: | Admin User / User press terminate device button. | |
| Post-conditions: | Admin User terminates a device successfully. | |
| Flow of events: | **User Action** | **System Action** |
|  | NULL |  |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | Calculate calories | |
| Actors: | User | |
| Pre-conditions: | Type of food exists. | |
| Post-conditions: | System calculate the calories gained. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enters the meal name. |  |
|  | 2- System checks the database. |
|  |  | 3-System calculate calories. |
| Exceptions: | **User Action** | **System Action** |
|  | 1-User enters the meal name. |  |
|  |  | 2-System didn’t find the food at the database. |
|  |  | 3-ask the user to enter the food |
|  | 4-User add the food. |  |
|  |  | 5-system calculate calories. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | Add food | |
| Actors: | User | |
| Pre-conditions: | User enter the name of the meals he eats. | |
| Post-conditions: | Meal name is added. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User adds the meal name. |  |
|  | 2- System adds the meal name to the database. |
| Exceptions: | **User Action** | **System Action** |
|  | 1-User adds the meal name. |  |
|  |  | 2-System checks database |
|  |  | 3-Meal name exists |
|  |  | 4-Asks the user to add another name. |
|  | 5-User adds another name. |  |
|  |  | 6-meal name is added. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | Health card | |
| Actors: | User | |
| Pre-conditions: | User is signed in. | |
| Post-conditions: | Shows health card information. | |
| Flow of events: | **User Action** | **System Action** |
| 1-User chooses type of information needed. |  |
|  | 2-System displays information chosen. |
| Exceptions: | **User Action** | **System Action** |
|  |  | 1-Health card is empty. |
|  |  | 2-System displays a blank page. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 13 | |
| Use Case Name: | Edit User | |
| Actors: | Administrator | |
| Pre-conditions: | Administrator chooses a user to be edited. | |
| Post-conditions: | User Info is updated. | |
| Flow of events: | **User Action** | **System Action** |
| 1-Admin Press chooses a user account | 3 – System displays User info |
| 2 – Admin clicks on edit user button  4 – Admin edits the user info | 5 – System displays successful editing notification |
| Exceptions: | **User Action** | **System Action** |
|  | NULL |  |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | 14 | |
| Use Case Name: | Enable Notification | |
| Actors: | Admin-User/Home-User | |
| Pre-conditions: | User clicks on Edit Notifications | |
| Post-conditions: | User successfully enabled the desire notification.  User returns to home page. | |
| Flow of events: | **User Action** | **System Action** |
| 1-User launches application |  |
|  | 2- System displays home page to user |
| 3- User clicks on Settings |  |
| 4- User clicks on Edit Notifications |  |
|  | 5-System displays the connected devices. |
| 6-User chooses the desired device and enables the notification button. |  |
|  |  | 7-System notifies the user that notifications have been enabled to the selected device. |
| Exceptions: | **User Action** | **System Action** |
|  |  |
|  | NULL |  |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | 15 | |
| Use Case Name: | Disable Notification | |
| Actors: | Admin-User/Home-User | |
| Pre-conditions: | User clicks on View Settings | |
| Post-conditions: | User clicked on Edit Notifications  User returns to home page. | |
| Flow of events: | **User Action** | **System Action** |
| 1-User launches application |  |
|  | 2- System displays home page to user |
| 3-User clicks on settings. |  |
| 4- User clicks on Edit Notifications |  |
|  | 5-System displays the connected devices. |
| 6-User chooses the desired device and disables the notification button. |  |
|  |  | 7- System notifies the user that notifications have been disabled to the selected device. |
| Exceptions: | **User Action** | **System Action** |
| NULL |  |
|  | NULL |  |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 16 | |
| Use Case Name: | Add User | |
| Actors: | Admin User | |
| Pre-conditions: | Admin user clicks Edit Users | |
| Post-conditions: | Admin user successfully added a new user.  Admin user returns to home page. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User launches the application. |  |
|  | 2- System displays the home page to the user. |
| 3- User clicks on settings and then Edit Users option and chooses to add a user. |  |
|  | System displays the search box and waits for user to enter the email connected with the other user he wants to add. |
| 5- User enters the desired email address. |  |
|  | 6- System searches for the entered email address and displays the user if found. |
|  | 7-User chooses to add the user to his home board. |  |
|  |  | 8- System notifies the admin user that a new user was successfully added. |
| Exceptions: | **User Action** | **System Action** |
| 1-User enters an email address that is not found or invalid. |  |
|  | 2- System displays an error message. |
|  |  | 3- System asks the user to enter a valid username. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 17 | |
| Use Case Name: | Remove User | |
| Actors: | Admin User | |
| Pre-conditions: | User clicks on Edit Users | |
| Post-conditions: | Admin user successfully removed a user.  Admin user returns to the home page. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User launches the application. |  |
|  | 2- System displays the home page to the user. |
| 3- User clicks on settings then on Edit Users option and chooses to remove a user. |  |
|  | 4- System displays connected users. |
| 5- User chooses the desired user to be removed. |  |
|  |  | 6-System removes the chosen user and notifies the admin user that a user was successfully removed. |
| Exceptions: | **User Action** | **System Action** |
| 1- User chooses to remove a user and there are no users connected to this board. |  |
|  | 2- System displays that there are no users connected. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

# Ownership Report

* **Remove the following notes and any red notes**
* **For every item in this document, write the owners. If someone is owner of something, s/he understands it 100%**
* **Team leader must verify the table with the team members.**

|  |  |
| --- | --- |
| **Item** | **Owners** |
|  |  |
|  |  |