Software Requirements Specification

for

Software Development

Version 1.0 approved

Prepared by Gonul Kilic

Surabhi Trivedi

Edward Ai

Pooja Khari

UMASS Boston

02.16.2019

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 Software Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

The goal of this document is to provide support information on the Software Development project. It will attempt to explain the functionality of the project and its features. This project seeks to find database management solutions for large buildings like various hotels. The company is already using Google Forms to collect data and analyze them. The solution will replace this system with a web application providing a customized view and identification. The activities of Administrator, Manager and Technician with Google Forms will be done through a web-based database solution that provides more customized and secure features.

## Intended Audience and Reading Suggestions

This Software Requirements document is intended for:

− Developers to review project’s features and more easily understand where their efforts should be targeted to improve or add more features to this project.

− Project testers can use this document to prepare their test cases. Also, it is easier to find out probable bugs with getting help from requirement document.

− Users can read this document to learn about the capabilities of the project.

## Product Scope

This project seeks to find database management solutions for large buildings like the hotels. The company is currently using Google Forms to collect data and analyze them. The solution will replace this system with web-based application. This replacement will provide easier access to view and identification to the data. The distribution of roles and responsibilities for Administrator, Manager and Technician with the new solution will be the same as in Google Forms. The new database solution aims to be more advance and secure than Google Forms. The biggest complaint for the company about Google Forms is struggling with working in a disorganized and chaotic environment so this project aims to satisfy the company with supplying ordinate working environment.

## References

The access to the company’s Google Forms and the system documents are taken as a reference.

# Overall Description

## Product Perspective

This product allows users to login and keep record the data which Administrator needs to know for the maintenance of the building. Every user will have different level of authorization based on their status. All the recorded information will be able to view from all users whenever needed.

## Product Functions

Product should show functionalities listed below:

* Administrator, Manager and Technician should be able to login to the system with their username and password.
* Administrator can approve registration of Manager and Technician.
* Manager can edit and Technician can fill forms and templates which is created by Administrator.
* The following forms should be provided to the users:
* Project Tracker
* Pool Readings
* Chiller Log Sheet
* B-2 Level Rounds and Readings
* Dayshift Lighting Rounds
* Dayshift Rounds
* When the user clicks on the forms, the information needed from Administrator should be there on the screen.
* System should provide multiple users and corresponding access control levels within each form and the entire database.
* Every user should be able to create a report to view all records based on their authorization level.
* System should be able to export reports in different formats that depends on user selection like Word, PDF, Excel format.
* Users who are going to make a choice in each facility should be warned if the information is out of desired value.
* If there is a poor or no service then provided data should be held in cache and once service is restored then it is submitted automatically.
* System should keep logbook for each user with all changes made by user.
* System should confirm with user if they want to save the changes.
* System should catch repeating problems and flag them.
* User should be warned by system if the value they used is out of range.

## User Classes and Characteristics

All users who have basic experience with computer can be able to use this web application effectively as long as proper maintenance is provided.

Anyone who wants to understand the structure and capability of the project can read this document and has basic information about it.

## Operating Environment

The web application can be opened using any of the browser i.e. Google Chrome, Safari, Mozilla Firefox, Internet Explorer.

## Design and Implementation Constraints

* All the resources will be supplied from the company.
* After delivering the web application, company will be need software maintenance.
* All data and information related company operation would be protected.

## User Documentation

User documentation which includes User Manual will be hand in with the project to the company.

## Assumptions and Dependencies

# External Interface Requirements

## Software Interfaces

The Workflow of the Application consists of three roles Admin, Manager and Technician. Each role will have access to specific activity of the form. For example, Admin will only be able to create a new form exclusively.

# System Features

The features of application have shown with the use case diagrams as below with the subtitles.

## Login Feature

|  |  |
| --- | --- |
| Use Case | Login to the System. |
| Description | Allows anyone who wants to login system. |
| Actors | Administrator, Manager, Technician |
| Precondition | User should have an internet access. |
| Post condition | User should be able to login system. |
| Steps | If user already has an account he/she should enter her/his email and password information and then click to Login button.  * If user doesn’t have an account he/she should register for the account by clicking register button and should wait for an approval from Administrator or Manager. * After getting an approval from the authorized person this person should be able to login with his/her login information. |

## View List of Building

|  |  |
| --- | --- |
| Use Case | View List of Building |
| Description | When the user login to the system they should be able to see list of buildings based on their authorization level and responsibilities. |
| Actors | Administrator, Manager, Technician. |
| Precondition | User should have an internet access and be able to login system. |
| Post condition | User should be able to access the system. |
| Steps | After user login to the system they should be able to see the list of building.  * Administrator should see the list of all buildings. * Manager and Technician should see the buildings that he/she has responsible for it. |

## Assigning a Roles

|  |  |
| --- | --- |
| Use Case | Assigning a Roles |
| Description | In the system based on their authorization users should be able to assign task and roles to the other users. |
| Actors | Administrator, Manager |
| Precondition | User should have an internet access and be able to login system. The needed authorization for the users should be given by Admin for being able to assign roles. |
| Post condition | After user assigned the role to the other user this user should be able to have all authorization for doing his/her job. |
| Steps | Admin should be able to define Administrator to the system.  * Administrator should be able to assign building and task to the Manager and Technician. * Manager should be able to assign building and task to the Technician. |

## Create a New Building

|  |  |
| --- | --- |
| Use Case | Create a New Building |
| Description | The application should be flexible with the way whenever Admin or Administrator wants to use it for another building he should be able to add new building. |
| Actors | Admin, Administrator |
| Precondition | User should be Admin or Administrator. |
| Post condition | New Building should be added to the system with all the feature that any other building has. |
| Steps | Administrator should use the button on his/her screen to create a new building.  * Once he clicks on the button new building should be created on database too and he/she should be able to see the building with all forms structure. * Admin should be able to change the structure of the forms. |

## View Data

|  |  |
| --- | --- |
| Use Case | View Data |
| Description | All users should be able to see all data which have been put in the form. |
| Actors | Administrator, Manager, Technician |
| Precondition | User should have an authorization to login system. |
| Post condition | After selecting to view a data user should be able to see on her/his screen all the needed data. |
| Steps | * User should select to view a data based on some criteria like form type, time, user name etc. * After selecting on criteria user should click to View Report button. * After clicking on View Report button all information should be shown on the screen to the user. |

## Edit Form Content

|  |  |
| --- | --- |
| Use Case | Edit form contents |
| Description | Administrator and Manager should be able to change with the form content which Technician fill in the form. |
| Actors | Administrator, Manager |
| Precondition | Users should be Administrator or Manager to be able to make this change. |
| Post condition | Form should be updated with the new content. |
| Steps | * Administrator Manager should open the form. * After open the form they should click on Edit button. * Then they should be able to make a change. * After making all the changes they should click on the Submit button to save the changes on the form. |

## Create a New Form

|  |  |
| --- | --- |
| Use Case | Creating a new form |
| Description | The needs related the building can be changed and the Administrator should be able to create a new type of form from the scratch. |
| Actors | Administrator |
| Precondition | User should be an Administrator to be able to make this change. |
| Post condition | New form should be added to the system and can be shown in the form list under the building information. |
| Steps | * Administrator should click on the New Form button for creating a new form. * Admin should be able to create a question and multiple-choice answers. * When admin complete the creation of the form he/she should click on the Submit button. * And form should be created. |

## Fill the Data

|  |  |
| --- | --- |
| Use Case | Filling an information in the form |
| Description | All users should be able to fill the forms with needed information from Administrator. |
| Actors | Administrator, Manager, Technician |
| Precondition | User should be able to login to the system. |
| Post condition | After filling and submission of the form they should be able to submit and view the form. |
| Steps | * User should choose the type of the form from the screen. * But desired information in the form. * After filling the form should click on button to submit the form. * Form should be submitted. |

## Create and Export the Report

|  |  |
| --- | --- |
| Use Case | Create and exporting the report |
| Description | All users should be able to create a report and export them in their computer with the desired format like Word, PDF, Excel. |
| Actors | Administrator, Manager, Technician |
| Precondition | User should have an access to the system. |
| Post condition | Report should be created on the screen or exported in their computer based on their chose. |
| Steps | * User should click Create Report button. * User should select from desired values. * After specified the selection he/she should clicked on View report button. * After click on View Report button the table with the desired information should be shown to the user. * User should be able to export report on his/her computer with clicking on Export Report button. * After Export Report button user should be asked in which format they want their report. * After users selection the report should be created and downloaded in users machine. |

## Keeping a Record in Cache

|  |  |
| --- | --- |
| Use Case | Keeping a record in cache |
| Description | As sometimes Technician should fill the form at the basement of the building it is possible to have a low connection or disconnection while they are doing their job. So, system should prevent this problem with keeping record in the cache and Submit them automatically once the connection gets better. |
| Actors | System |
| Precondition | User should fill the form and click on the Submit button. |
| Post condition | Form should be submitted after connection problem is solved. |
| Steps | * After submission of the form system should check the network connection. * If there is no connection or not enough connection system should keep the records in the cache. * Once the connection gets better the recorded form in the cache should be sent to database. |

## Keeping a Logbook

|  |  |
| --- | --- |
| Use Case | Keeping a logbook |
| Description | All the actions from all users like making. Change, login to the system, creating a form, update the form etc. should be recorded in the logbook |
| Actors | System |
| Precondition | Users should have an action on the system. |
| Post condition | Every action from the users should be recorded |
| Steps | * Whatever changes are done by users should be recorder with their profile, data and time information. * Also, login information should be recorded too. |

## Determine Repeated Problems

|  |  |
| --- | --- |
| Use Case | Determine repeated problems |
| Description | System should be able to keep the records of problems and whenever the problem is repeated system should warn the user about this situation. |
| Actors | System |
| Precondition | The previous problems should be recorded.  If the same problem happened system should determine this situation. |
| Post condition | User should receive a warning from the system. |
| Steps | * User should fill the information. * If there is any problem system should check the record if there is any same repeated problem. * If system find out the same problem from the records should show warning on the screen. |

# Other Nonfunctional Requirements

## Performance Requirements

* System should have cache besides of database.
* System should check the records for each error to detect if this error is repeated or not.
* System should be able to export selected data from database to users’ machine.
* Users should be able to reach to system with any browser.
* System should keep logbook for security reason.
* System should have security feature with user email and password control.

## Safety Requirements

* Users can not make any change in database or system structure.
* There should be backup for the database for just in case there are some crushes on database system.

## Security Requirements

Security was taken as a very serious nonfunctional requirement so that the data and information of each user is protected from visibility and possible alteration by the other users.

Thus, the system provides a user account which enables users to enter their respective profile with using a login id and password mechanism.

Also, the authorization level between users should be identified and logbook should be kept for security reason.

## Software Quality Attributes

* Software should be run on every machine which has an internet access.
* The structure of software should be fixed except giving some flexibility to the Administrator depends on needs.
* As users are working on Google Forms the software should be as similar as possible to the Google Forms working process to keep software user friendly. The user should feel differences just with more comfort when they are using an application.
* Application should be very clear to understand and use it for every level of user.

## Business Rules

* Administrator should be only one who can create a building.
* Administrator and Manager should be the only one can make a change in the form.
* Admin should be the only one can assign administrator to the system.
* Administrator and manager should be the ones who can reach logbook records.
* Administrator should be only one who can assign manager for the buildings.