Software Requirements Specification

for

FOES Data Management System

Version 5.0

Prepared by

Lee Chian Hui, Ng Jing Ru, James Wong Siew Huei, Yeum Kyu Seok

Curtin University Malaysia

8 October 2022

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
All Members	16/3/2022	Initial Draft	1.0
Ng Jing Ru, James Wong Siew Huei	13/4/2022	Changes based on the feedback, added meeting minutes, added chapter 2.5 and 2.7	2.0
All Members	6/5/2022	Changes the format, added UI, added meeting minutes	3.0
All Members	24/5/2022	System Features, UI changes, added meeting minutes	4.0
All Members	8/10/2022	System Features, removed agile section, changed technology section	5.0

1. Introduction

Contributor(s): Kyu Seok yeum

1.1 Purpose

The purpose of the "FOES Data Management System" project is to help the Faculty of Engineering and Science department to implement a database management application. The database will store information of FOES such as, staff, activities, assets, orders etc. The application will contain functionalities requested by the stake holder which includes, automatic data insertions using excel sheet, and generation of reports from the application.

The purpose of the document is to provide the analyzed criteria and specifications required for the project. This document is designed to help with better understanding of the project for both stakeholder and the project participants. The SRS document contains a detailed overview of the software design architecture, project's requirements and other information that are related to the project.

1.2 Document Conventions

Bold words represent: Emphasize on correct terms.
Italic words represent: Guideline or Reference & Citations.
Highlights represent: Any updates from the previous version of the document.

1.3 Intended Audience and Reading Suggestions

The Intended Audience of this document for both stakeholders and the project participants are listed below. The field knowledge is not required for the reading of this document, however the background knowledge for Computer Science will help with the understanding of the project.
It is recommended to read the document in sequential order, however if the reader wants to read only domain specific information, the suggestions are as below.

Sections:
1. Introduction
2. Overall Description
3. System features
4. External Interface Requirements
5. Non-Functional Requirements
6. Other Requirements

Audiences:	Client & Stakeholder: 1, 2, 3, 4, 5, 6
Software developer:	2, 3, 4, 5, 6
UI / UX Designer:	3, 4
Tester:	3, 4, 5
Document writer:	1, 3, 5, 6

** The sections are represented as number*

1.4 Project Scope

The Goal of the project is to implement a web-based database management application that will be managed by the Faculty of Engineering and Science department. The product will provide convenient features on managing the database system with user friendly interface and flexible customization features. The product is designed for ease of maintenance and specific functionality requested by the stakeholders such as automatic data insertion from spreadsheets are listed in the section 3. System Features, with detailed explanation.

1.5 References

IEEE. 1998. “IEEE Recommended Practice for Software Requirements Specifications.” *IEEE*, (Oct), 1-40. 10.1109/IEEESTD.1998.88286.

IEEE. n.d. “SRS Template.” Blackboard. Accessed March 13, 2022. https://lms.curtin.edu.au/bbcswebdav/pid-9806557-dt-content-rid-60845244_1/xid-60845244_1.

2. Overall Description

Contributor(s): James Siew Huei Wong, Jing Ru Ng

2.1 Product Perspective

FOES Data Management System is a new data management system which is expected to replace the existing manual data entry system. The proposed system will be used to keep track of all information of the staff, for example their personal details, achievements, and other features such as general reports.

2.2 Product Features

Administrative officers should be able to keep track of information related to all staff, outreach and community engagement activities, internal and external requests, store information, manage staff assets, customize and generate report templates, enter data manually and automatically from the system.

2.3 User Classes and Characteristics

Super Admins
- This user role has the ability to add, delete and alter any information in the data management system.
- Have the permission to add, delete, alter admins.

Admin
- This user role has the ability to add, delete and alter any information in the data management system.

2.4 Operating Environment

This system is expected to be used in a web environment on a desktop PC. There is no expectation for this web application to be used on any other form of devices.

2.5 Design and Implementation Constraints

Frontend – ReactJS
Backend – Laravel
Database – MySQL
Storage & Deployment – Curtin University's internal hardware.

2.6 User Documentation

A user manual and a documentation of the code will be delivered at the end of this project.

2.7 Assumptions and Dependencies

- Assume that there is only admin role needed in this program, there will not be any other user roles.

- A special role called super admin would be created for the sole purpose of the creation and delete of an admin account. This is for security purposes, as a normal admin role should not be able to create another new admin role.

- The user can only access the program while in Curtin University, as it would be hosted locally.

3. System Features

Contributor(s): Chian Hui Lee, Jing Ru Ng, Kyu Seok yeum, James Siew Huei Wong

3.1 Table Customization

3.1.1 Description and Priority

The user may manually change the structure of the database such as adding a new column to the table.

3.1.2 Stimulus/Response Sequences

Stimulus: The user clicks on the table that they would like to update on the navigation bar.

Respond: The table structure is changed based on the selection, and the application displays the updated table.

3.1.3 Functional Requirements

Frontend:

REQ-1: The user selects a table from the navigation bar.
REQ-2: The system displays the list of columns on that table.
REQ-3: The user may add, edit or delete the column of the table.
REQ-4: Display response to the user on success or failure.

Backend:

REQ-1: Validation if the requested change may be operated.
REQ-2: On successful update, update the database and send response 200 OK to the frontend.
REQ-3: On failure, send 400 Bad Request to the frontend.

3.2 Generation of Report

3.2.1 Description and Priority

Users may generate reports from the database. The report will contain the data on what the user has selected. It can generate either a PDF or CSV file.

3.2.2 Stimulus/Response Sequences

Stimulus: The user clicks on the “Export” or “Export all” button.

Respond: The system will generate a report based on the selection made by the user.

3.2.3 Functional Requirements

Frontend:

REQ-1: The user clicks the "Export" or “Export All” button.
REQ-2: A preview of the file will be shown to the user if it is a PDF. If it is an excel, it will ask the user on where they would like to save the file to.
REQ-3: The user can then download the file into their system.

Backend:

REQ-1: Load the requested data from the database and send it to the frontend in either JSON or XML format.

3.3 Import Data Automatically

3.3.1 Description and Priority

The application will automatically insert the data into the database from the input CSV.

3.3.2 Stimulus/Response Sequences

Stimulus: The user input a CSV file to the application.

Respond: The software will insert data into the database from the input CSV file.

3.3.3 Functional Requirements

Frontend:

REQ-1: User manually loads the spreadsheet file containing data.

REQ-2: Preview of data to be inserted should be displayed.

REQ-3: The confirmation on insertion should pop up.

REQ-4: The application shows an updated database.

Backend:

REQ-1: The system should perform data validation to check if any error will occur when data is inserted.

REQ-2: Upon succession, the backend application inputs the data into the database.

REQ-3: Upon failure, the system will send a 400 Bad Requests to the frontend.

3.4 Dashboard View Customization

3.4.1 Description and Priority

Admin can customize the view of the list in the dashboard however they see fit.

3.4.2 Stimulus/Response Sequences

Stimulus: The admin will have to click on the “Customize Column” button on the top right of each of the dashboard to customize that view.

Respond: The system will check if the view option is checked or not for that particular column and render the final result on the dashboard.

3.4.3 Functional Requirements

Frontend:

REQ-1: Data of the asset information should be displayed in a table.
REQ-2: The user should be able to enter and edit the data in the fields.
REQ-3: The system will show ‘Information has been created successfully.’ when the data validation has no issue and there is no duplicate account in the database.
REQ-4: The system will show ‘Information has been updated successfully.’ when the data validation has no issue.
REQ-5: The system will show ‘Information has been deleted successfully.’ when the user has confirmed to delete the data.
REQ-6: The user should be able to enter search term to search on certain data.
REQ-7: The system would show an error message on that particular field if it failed the validation check.
REQ-8: The data in research awards table and staff dashboard should be synchronized.

Backend:

REQ-1: The system should perform authorization checks before performing any actions for the research awards information.
REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.
REQ-3: The system should update the data in the database once the action is confirmed.
REQ-4: The system should not update the data if detects ineligible data.
REQ-5: The system should be able to return the request data upon the search term.

3.5 Data Backup & Restore

3.5.1 Description and Priority

The user may backup the database system or restore a previous database system.

3.5.2 Stimulus/Response Sequences

Stimulus: The user clicks on the “Backup” or “Restore” button after choosing the database.

Respond: The system saves a backup of the database or restore to a selected database.

3.5.3 Functional Requirements

Frontend:

REQ-1: The system should perform authorization checks before performing any actions.

REQ-2: The user selects a file to restore the database, clicks the “Restore” button and the system restores the previous database system.

REQ-3: On any successful operation on backup and restore, the system prompts a success message.

REQ-4: On any failure, system prompt error message to the user.

Backend:

REQ-1: The system should perform authorization checks before performing any actions.

REQ-2: The system should save database backup files on backup request.

REQ-3: The system should restore the selected database on restore request.

3.6 Research Awards Information

3.6.1 Description and Priority

The user should be able to create, read, update, delete and search on the research awards information.

3.6.2 Stimulus/Response Sequences

Stimulus: The user clicks on the research awards dashboard. Then they can either choose the create, update, delete or search button.

Respond: - The system will generate a new research awards information.

- The system will display all of the research awards information in the form of a table.

- The system will update the database once an entry has been updated.

- A successful response will be shown once the data is deleted from the database.

- The system will return the searched data based on the search term.

3.6.3 Functional Requirements

Frontend:

REQ-1: Data of the asset information should be displayed in a table.
REQ-2: The user should be able to enter and edit the data in the fields.
REQ-3: The system will show ‘Information has been created successfully.’ when the data validation has no issue and there is no duplicate account in the database.
REQ-4: The system will show ‘Information has been updated successfully.’ when the data validation has no issue.
REQ-5: The system will show ‘Information has been deleted successfully.’ when the user has confirmed to delete the data.
REQ-6: The user should be able to enter search term to search on certain data.
REQ-7: The system would show an error message on that particular field if it failed the validation check.
REQ-8: The data in research awards table and staff dashboard should be synchronized.

Backend:

REQ-1: The system should perform authorization checks before performing any actions for the research awards information.
REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.
REQ-3: The system should update the data in the database once the action is confirmed.
REQ-4: The system should not update the data if detects ineligible data.
REQ-5: The system should be able to return the request data upon the search

term.

REQ-6: The system should update the data in staff dashboard once the user has performed any action on the research awards table.

3.7 Outreach and Community Engagement Activities

3.7.1 Description and Priority

The user should be able to create, read, update, delete and search on the outreach and community engagement activities information.

3.7.2 Stimulus/Response Sequences

Stimulus: The user clicks on the outreach and community engagement activities dashboard. Then they can either choose the create, update, delete or search button.

Respond: - The system will generate a new outreach and community engagement activity information.

- The system will display all of the outreach and community engagement activity information in the form of a table.

- The system will update the database once an entry has been updated.

- A successful response will be shown once the data is deleted from the database.

- The system will return the searched data based on the search term.

3.7.3 Functional Requirements

Frontend:

REQ-1: Data of the outreach and community engagement activity should be displayed in a table.

REQ-2: The user should be able to enter and edit the data in the fields.

REQ-3: The system will show ‘Information has been created successfully.’ when the data validation has no issue and there is no duplicate account in the database.

REQ-4: The system will show ‘Information has been updated successfully.’ when the data validation has no issue.

REQ-5: The system will show ‘Information has been deleted successfully.’ when the user has confirmed to delete the data.

REQ-6: The user should be able to enter search term to search on certain data.

REQ-7: The system would show an error message on that particular field if it failed the validation check.

REQ-8: The user should be able to choose the category of the outreach and community engagement activity.

Backend:

REQ-1: The system should perform authorization checks before performing

any actions for the activity information.

REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.

REQ-3: The system should update the data in the database once the action is confirmed.

REQ-4: The system should not update the data if detects ineligible data.

REQ-5: The system should be able to return the request data upon the search term.

REQ-6: The system should be able to store the information according to the category that the user selected on.

3.8 MOU & MOA Program Information

3.8.1 Description and Priority

The user should be able to create, read, update, delete and search on the MOU & MOA program information.

3.8.2 Stimulus/Response Sequences

Stimulus: The user clicks on the MOU & MOA program activities dashboard. Then they can either choose the create, update, delete or search button.

Respond: - The system will generate a new MOU & MOA program activity information.

- The system will display all of the MOU & MOA program activity information in the form of a table.

- The system will update the database once an entry has been updated.

- A successful response will be shown once the data is deleted from the database.

- The system will return the searched data based on the search term.

3.8.3 Functional Requirements

Frontend:

REQ-1: Data of the MOU & MOA program activity should be displayed in a table.

REQ-2: The user should be able to enter and edit the data in the fields.

REQ-3: The system will show ‘Information has been created successfully.’ when the data validation has no issue and there is no duplicate account in the database.

REQ-4: The system will show ‘Information has been updated successfully.’ when the data validation has no issue.

REQ-5: The system will show ‘Information has been deleted successfully.’ when the user has confirmed to delete the data.

REQ-6: The user should be able to enter search term to search on certain data.

REQ-7: The system would show an error message on that particular field if it

failed the validation check.

REQ-8: The user should be able to choose the category of the MOU & MOA program activity.

Backend:

REQ-1: The system should perform authorization checks before performing any actions for the activity information.

REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.

REQ-3: The system should update the data in the database once the action is confirmed.

REQ-4: The system should not update the data if detects ineligible data.

REQ-5: The system should be able to return the request data upon the search term.

REQ-6: The system should be able to store the information according to the category that the user selected on.

3.9 Admin Account Information

3.9.1 Description and Priority

Super admin should be able to create, read, update, delete and search an admin account. Admin should be able to read and search only.

3.9.2 Stimulus/Response Sequences

Stimulus: The super admin clicks on the admin dashboard. Then they can either choose the create, update, delete or search button.

Respond: - The system will generate a new admin account.

- The system will display all of the admin accounts in the form of a table.

- The system will update the database once an entry has been updated.

- A successful response will be shown once the data is deleted from the database.

- The system will return the searched data based on the search term.

3.9.3 Functional Requirements

Frontend:

REQ-1: Data of the admin accounts should be displayed in a table.

REQ-2: The super admin and the admin should be able to enter and edit the data in the fields.

REQ-3: The system will show ‘Account has been created successfully.’ when the data validation has no issue and there is no duplicate account in the database.

REQ-4: The system will show ‘Information has been updated successfully.’

when the data validation has no issue.

REQ-5: The system will show ‘Account has been deleted successfully.’ when the super admin has confirmed to delete the data.

REQ-6: The super admin and the admin should be able to enter search term to search on certain data.

REQ-7: The system would show an error message on that particular field if it failed the validation check.

Backend:

REQ-1: The system should perform authorization checks before performing any actions for the admin account.

REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.

REQ-3: The system should update the data in the database once the action is confirmed.

REQ-4: The system should not update the data if detects ineligible data.

REQ-5: The system should be able to return the request data upon the search term.

3.10 Staff Information

3.10.1 Description and Priority

The user should be able to create, read, update, delete and search on the asset information.

3.10.2 Stimulus/Response Sequences

Stimulus: The user clicks on the staff dashboard. Then they can either choose the create, update, delete or search button.

Respond: - The system will generate a new staff information.

- The system will display all the staff information in the form of a table.

- The system will update the database once an entry has been updated.

- A successful response will be shown once the data is deleted from the database.

- The system will return the searched data based on the search term.

3.10.3 Functional Requirements

Frontend:

REQ-1: Data of the staff information should be displayed in a table.

REQ-2: The user should be able to enter and edit the data in the fields.

REQ-3: The system will show ‘Staff information has been created successfully.’ when the data validation has no issue and there is no duplicate account in the database.

REQ-4: The system will show ‘Information has been updated successfully.’

when the data validation has no issue.

REQ-5: The system will show ‘Staff information has been deleted successfully.’ when the user has confirmed to delete the data.

REQ-6: The user should be able to enter search term to search on certain data.

REQ-7: The system would show an error message on that particular field if it failed the validation check.

REQ-8: The user should be able to upload attachment for certain fields.

REQ-9: Research Awards of the staff should also be displayed under the staff data.

Backend:

REQ-1: The system should perform authorization checks before performing any actions for the staff information.

REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.

REQ-3: The system should update the data in the database once the action is confirmed.

REQ-4: The system should not update the data if detects ineligible data.

REQ-5: The system should be able to return the request data upon the search term.

REQ-6: The system should be able to store multiple attachment files.

REQ-7: The data of Staff Dashboard and Research Awards should be connected.

3.11 Asset Information

3.11.1 Description and Priority

The user should be able to create, read, update, delete and search on the asset information.

3.11.2 Stimulus/Response Sequences

Stimulus: The user clicks on the asset dashboard. Then they can either choose the create, update, delete or search button.

Respond: - The system will generate a new asset information.

- The system will display all the asset information in the form of a table.

- The system will update the database once an entry has been updated.

- A successful response will be shown once the data is deleted from the database.

- The system will return the searched data based on the search term.

3.11.3 Functional Requirements

Frontend:

REQ-1: Data of the asset information should be displayed in a table.

REQ-2: The user should be able to enter and edit the data in the fields.

REQ-3: The system will show ‘Asset information has been created successfully.’ when the data validation has no issue and there is no duplicate account in the database.

REQ-4: The system will show ‘Information has been updated successfully.’ when the data validation has no issue.

REQ-5: The system will show ‘Asset information has been deleted successfully.’ when the user has confirmed to delete the data.

REQ-6: The user should be able to enter search term to search on certain data.

REQ-7: The system would show an error message on that particular field if it failed the validation check.

Backend:

REQ-1: The system should perform authorization checks before performing any actions for the asset information.

REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.

REQ-3: The system should update the data in the database once the action is confirmed.

REQ-4: The system should not update the data if detects ineligible data.

REQ-5: The system should be able to return the request data upon the search term.

3.12 Mobility Activities Information

3.12.1 Description and Priority

The user may create, read, update, delete and search the mobility of staff and student.

3.12.2 Stimulus/Response Sequences

Stimulus: The user clicks on the Mobility tab in the dashboard. Then the user may create, update, delete or search the mobility data.

Respond: - The system will generate new Mobility data.

- The system will display all the Mobility data in the table.

- The system will update the database once an entry has been updated.

- A successful response will be shown once the data is deleted from the database.

3.12.3 Functional Requirements

Frontend:

REQ-1: Table of mobility data is displayed to the application3

REQ-2: The system displays “Create Activity” button where user may create new mobility data

REQ-3: The system displays update button where user may update an initial data

REQ-4: The system displays delete button where user may delete a data

REQ-5: The system displays “Staff” and “Student” database under mobility

REQ-6: The system displays “Inbound” and “Outbound” tab under each Staff and Student database

REQ-7: On any successful operation on create & update, system prompt success message

REQ-8: On any failure, system prompt error message to the user.

Backend:

REQ-1: The system should perform authorization checks before performing any actions.

REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.

REQ-3: The system should update the data in the database once the action is confirmed

REQ-4: The system should not update the data if it detects ineligible data.

REQ-5: The system should update the data Mobility table once the user has performed any action on the research award table.

3.13 Login feature

3.13.1 Description and Priority

The user can login to the application.

3.13.2 Stimulus/Response Sequences

Stimulus: The user will be greeted by the login page when they first visit the web page.

Respond: The user will be able to log into the application with the correct credentials.

3.13.3 Functional Requirements

Frontend:

REQ-1: Login page will be shown to the user.

REQ-2: The user has to enter their credentials.

REQ-3: The system displays the dashboard of the webpage upon successful login.

REQ-4: On any failure, the system will prompt error message to the user.

Backend:

REQ-1: The system should perform authorization checks before performing any actions.

REQ-2: The system should perform data validation and check for duplicate records when creating and updating the data.

REQ-3: The system will check if the email and password is valid or not. If either one of them is not valid, an error message will be shown.

REQ-4: Once valid, a Json Web Token will be generated for authentication purposes.

REQ-5: On any failure, the system will prompt error message to the user.

4. External Interface Requirements

Contributor(s): Chian Hui Lee

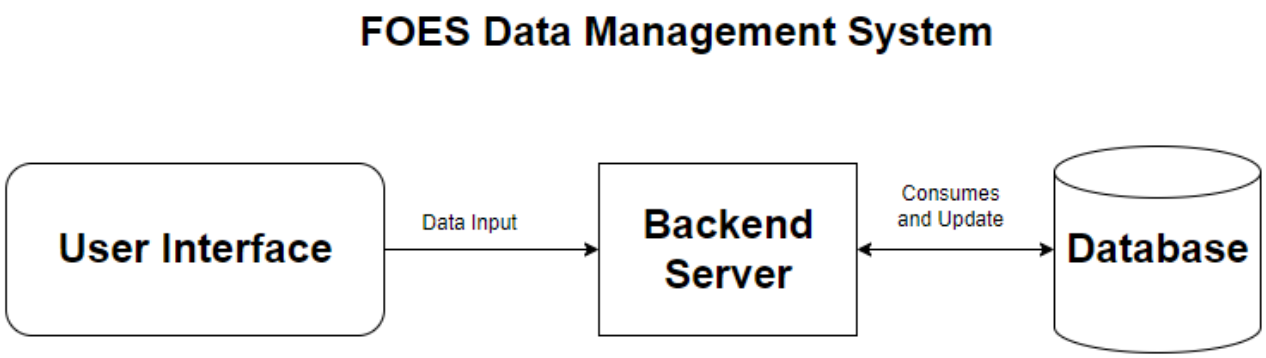
4.1 User Interfaces

Front-end user interface: The user interface can be found here

4.2 Hardware Interfaces

Not Applicable.

4.3 Software Interfaces



4.3.1 User Interface

Action will be requested from the front-end. Data will be passed from the front-end to the backend.

4.3.2 Backend Server

The backend server will handle requests from the frontend and communicate between the database and the user interface. Data validation, logic and the integration of the front-end will be done in the backend. The web services and APIs will also be done on the backend.

4.3.3 Database

To save all the data and tables. Example: User Table, Staff Table, Staff Asset Table and Room Table

4.4 Communications Interfaces

HTTPS communication standards will be used to protect the integrity and confidentiality of data between the user's computer and the site. The data will be encrypted and cannot be modified or corrupted during transmission. It also protects against man-in-the-middle attacks and builds user trust.

5. Other Nonfunctional Requirements

Contributor(s): Chian Hui Lee

5.1 Performance Requirements

The system must support multiple users online simultaneously.

5.2 Safety Requirements

Not applicable.

5.3 Security Requirements

For this system, the security requirements will include, but not limited to:

- Only super admin and admins can create, read, update, and delete staff's information.
- Only super admin can create, read, update, and delete admin's data.
- Apply password protection.

5.4 Software Quality Attributes

5.4.1 Reliability

The website should be reliable. It should be available at any time of the week. It should provide proper error handling. The website should not crash when error occurs.

5.4.2 Usability

The website should be user friendly. UI should be clear and easy to understand, so the user can perform the task efficiently and effectively. The website should not have redundant steps while performing action.

5.4.3 Manageability

The admin and super admin can manage the data by adding, editing, and deleting the staff's data information.

5.4.4 Correctness

The website should retrieve requests from users and display the data correctly. The website should be recording data entered by the user into the database correctly.

5.5 Business Rules

- The admin and super admin should be able to modify the data once he or she has logged in.
- Only the super admin should be able to manage admin's data.

6. Other Requirement

Contributor(s):

Not Applicable

Appendix A: Glossary

Not Applicable

Appendix B: Analysis Models

Not Applicable

Appendix C: Issues List

Not Applicable