# Software Requirements Specification for Software Engineering: Document Management System

Team 15, SyncMaster
Kyle D'Souza
Mitchell Hynes
Richard Fan
Akshit Gulia
Rafeed Iqbal

October 4, 2024

# Contents

# Revision History

| Date | Version | Notes |
|---|---|---|
| Date 1 | 1.0 | Notes |
| Date 2 | 1.1 | Notes |

# 1 Purpose of the Project

## 1.1 User Business

*Insert your content here.*

## 1.2 Goals of the Project

*Insert your content here.*

# 2 Stakeholders

## 2.1 Client

*Insert your content here.*

## 2.2 Customer

*Insert your content here.*

## 2.3 Other Stakeholders

*Insert your content here.*

## 2.4 Hands-On Users of the Project

*Insert your content here.*

## 2.5 Personas

*Insert your content here.*

## 2.6 Priorities Assigned to Users

*Insert your content here.*

### 2.7 User Participation

*Insert your content here.*

### 2.8 Maintenance Users and Service Technicians

*Insert your content here.*

## 3 Mandated Constraints

### 3.1 Solution Constraints

*Insert your content here.*

### 3.2 Implementation Environment of the Current System

*Insert your content here.*

### 3.3 Partner or Collaborative Applications

*Insert your content here.*

### 3.4 Off-the-Shelf Software

*Insert your content here.*

### 3.5 Anticipated Workplace Environment

*Insert your content here.*

### 3.6 Schedule Constraints

*Insert your content here.*

### 3.7 Budget Constraints

*Insert your content here.*

### 3.8 Enterprise Constraints

*Insert your content here.*

## 4 Naming Conventions and Terminology

### 4.1 Glossary of All Terms, Including Acronyms, Used by Stakeholders involved in the Project

*Insert your content here.*

## 5 Relevant Facts And Assumptions

### 5.1 Relevant Facts

*Insert your content here.*

### 5.2 Business Rules

*Insert your content here.*

### 5.3 Assumptions

*Insert your content here.*

## 6 The Scope of the Work

### 6.1 The Current Situation

*Insert your content here.*

### 6.2 The Context of the Work

*Insert your content here.*

### 6.3 Work Partitioning

*Insert your content here.*

### 6.4 Specifying a Business Use Case (BUC)

*Insert your content here.*

## 7 Business Data Model and Data Dictionary

### 7.1 Business Data Model

*Insert your content here.*

### 7.2 Data Dictionary

*Insert your content here.*

## 8 The Scope of the Product

### 8.1 Product Boundary

*Insert your content here.*

### 8.2 Product Use Case Table

*Insert your content here.*

### 8.3 Individual Product Use Cases (PUC's)

*Insert your content here.*

## 9 Functional Requirements

### 9.1 Functional Requirements

*Insert your content here.*

# 10 Look and Feel Requirements

## 10.1 Appearance Requirements

*Insert your content here.*

## 10.2 Style Requirements

*Insert your content here.*

# 11 Usability and Humanity Requirements

## 11.1 Ease of Use Requirements

*Insert your content here.*

## 11.2 Personalization and Internationalization Requirements

*Insert your content here.*

## 11.3 Learning Requirements

*Insert your content here.*

## 11.4 Understandability and Politeness Requirements

*Insert your content here.*

## 11.5 Accessibility Requirements

*Insert your content here.*

# 12 Performance Requirements

## 12.1 Speed and Latency Requirements

*Insert your content here.*

### 12.2 Safety-Critical Requirements

*Insert your content here.*

### 12.3 Precision or Accuracy Requirements

*Insert your content here.*

### 12.4 Robustness or Fault-Tolerance Requirements

*Insert your content here.*

### 12.5 Capacity Requirements

*Insert your content here.*

### 12.6 Scalability or Extensibility Requirements

*Insert your content here.*

### 12.7 Longevity Requirements

*Insert your content here.*

## 13 Operational and Environmental Requirements

### 13.1 Expected Physical Environment

*Insert your content here.*

### 13.2 Wider Environment Requirements

*Insert your content here.*

### 13.3 Requirements for Interfacing with Adjacent Systems

*Insert your content here.*

### 13.4 Productization Requirements

*Insert your content here.*

### 13.5 Release Requirements

*Insert your content here.*

## 14 Maintainability and Support Requirements

### 14.1 Maintenance Requirements

MS-MTN1. A deployment of the system should take no more than 30 minutes (not including testing, and building time).

MS-MTN2. The build time of the system should be no longer than 10 minutes (not including testing time).

MS-MTN3. All automated tests should be able to run in under 10 minutes

MS-MTN4. The system should have rigourous unit testing, line coverage should be $\geq 95\%$, branch coverage should be $\geq 90\%$.

MS-MTN5. All core functionalities of the system (i.e. Functional Requirements), should have both automated end-to-end and unit testing corresponding to them

MS-MTN6. The project must be able to be maintained by its users, as original developers will not be maintaining it after April 2, 2025.

### 14.2 Supportability Requirements

MS-SUP1. The application should have user-facing documentation on how to use the core functionalities of the system (i.e. functionalities described in functional requirements).

MS-SUP2. The application should have documentation for all API's for future maintainers.

MS-SUP3. The application should have documentation of internal functions and abstractions for future maintainers.

MS-SUP4. The application should have documentation on deployment, so users can deploy this application for themselves.

### 14.3 Adaptability Requirements

MS-ADP1. The application must be able to run on at least Google Chrome and Microsoft Edge browsers.

MS-ADP2. The application must be able to run on tablets, smartphones, and laptops.

MS-ADP3. The application must be able to run on Android, IOS, and Windows 10

## 15 Security Requirements

### 15.1 Access Requirements

*Insert your content here.*

### 15.2 Integrity Requirements

*Insert your content here.*

### 15.3 Privacy Requirements

*Insert your content here.*

### 15.4 Audit Requirements

*Insert your content here.*

### 15.5 Immunity Requirements

*Insert your content here.*

## 16 Cultural Requirements

### 16.1 Cultural Requirements

*Insert your content here.*

## 17 Compliance Requirements

### 17.1 Legal Requirements

*Insert your content here.*

### 17.2 Standards Compliance Requirements

*Insert your content here.*

## 18 Open Issues

*Insert your content here.*

## 19 Off-the-Shelf Solutions

### 19.1 Ready-Made Products

*Insert your content here.*

### 19.2 Reusable Components

*Insert your content here.*

### 19.3 Products That Can Be Copied

*Insert your content here.*

# 20 New Problems

## 20.1 Effects on the Current Environment

1. The application should not create redundant workload for tasks already carried out by existing solutions.

## 20.2 Effects on the Installed Systems

1. The application should not affect the system it is being run on other than taking in the required inputs and providing the appropriate outputs.

2. The application should not affect the systems it is interfacing with other than retrieving data, and sending data if necessary.

## 20.3 Potential User Problems

1. The user may not have access to the internet.

2. The user may not have a device which can run the application.

## 20.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

N/A

## 20.5 Follow-Up Problems

1. Business processes might change, changing the requirements of the application.

2. New software solutions may be introduced which make some features redundant.

3. Regulations may change adding or removing requirements.

# 21 Tasks

## 21.1 Project Planning

Project deliverables should be completed by the deadlines given in the course outline. GitHub will be used to track project milestones and tasks. Tasks will be assigned to individual team members or to groups. All work will be reviewed by other members of the team before being committed to the project. Feedback received from stakeholders, TAs, or the professor will be implemented in the project, and requirements will be changed accordingly.

Task 1. Set-up codebase and begin development of project.

Task 2. Work on documentation and deliverables.

Task 3. Get feedback from stakeholders, TAs, and the professor and implement suggested changes.

## 21.2 Planning of the Development Phases

1. *Proof of Concept*: Will start development after October 9th, 2024. Aim to complete by November 4th.
2. *Rev. 0*: Aim to complete by February 1st, 2024.
3. *Rev. 1*: Aim to complete by March 30th, 2024.
4. *Future revisions*: TBD

# 22 Migration to the New Product

## 22.1 Requirements for Migration to the New Product

*Insert your content here.*

## 22.2 Data That Has to be Modified or Translated for the New System

*Insert your content here.*

# 23 Costs

The cost for the application should not exceed $750 unless approved by the professor and the stakeholders for the project.

It is expected that the team will spend 40 man-hours per week on the project until its completion.

| Item | Cost | Description |
|---|---|---|
| Cloud Services | $ TBD | Amazon Web Services (AWS) |
| Domain Name | $ TBD | TBD |

# 24 User Documentation and Training

## 24.1 User Documentation Requirements

*Insert your content here.*

## 24.2 Training Requirements

*Insert your content here.*

# 25 Waiting Room

*Insert your content here.*

# 26 Ideas for Solution

*Insert your content here.*

# Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?