Silver Surfer

Software Requirements Specification (SRS) Document

Rough Draft

January 20, 2021

Silver Foxes

Faith Haiss, Amanda Hegidus, Ben Schroth, Marisa Stover

[Team Logo – it's so good we don't want to spoil it yet]

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Rough draft v1	Amanda Hegidus Faith Haiss Marisa Stover Ben Schroth	Rough draft first version. First time adding content to the document.	1/27/2021

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to outline the requirements and functions for the Silver Surfer project. The intended audience is the Silver Foxes group and the CS department faculty, who may be involved in the future.

1.2 Description

The software is being developed to provide professors who utilize the CS department Silver Server for their webpages. It will provide a working, usable template for professors to use so all pages on Silver will be functional and similar in layout. This will make it easier for students to find what they need, and it will be easier for professors to communicate with students.

1.3 Overview

This document will provide all of the initial requirements and conceptualization for the Silver Surfer project. After this initial description there will be specific details on the requirements that the team plans to include. Finally, the document will conclude with our planned Software Life Cycle Model and the reasoning behind this choice.

Commented [AH1]: done

1.4 Glossary of Terms

Terms:

Silver Server – The server used by the Mount Union Computer Science department.

Discord – A popular social network consisting of "servers" that allow users to

talk to other users via text and voice/video calls.

Abbreviations/Acronyms:

SLCM – (Software Life Cycle Model) A conceptual framework that allows the team to work on meeting project requirements efficiently and working on the maintenance of the project.

GUI – (Graphical User Interface) An interface that allows the user to interact with different visual elements.

UI – (User Interface) Includes the combination of human-computer interaction and the communication with a device.

VPN – (Virtual Private Network) Extends a private network over a public network and allows the user to send/receive information as if they were directly connected to the private network.

CS – Computer Science

UMU – University of Mount Union

1.5 Business Context

There is no direct external sponsor for this project. Upon completion, the Silver Surfer project may be adopted by the UMU CS department for their Silver Server webpages.

2 Overall Description

2.1 Product Perspective

This program will be a web-based interface that any user on either Mount Union's network or using the VPN will have the ability to check on class resources or communicate with their professors.

2.1.1 System Interfaces

Silver Surfer will be hosted on the silver.mountunion.edu server. Silver Surfer is a web application so it will be made available to any device with a browser.

- Hosting - The final application will be bundled in a docker container and installed on the silver servers.
- Next.js - The routing and API of Silver server to handle and serve requests to the application.
- SQL - The database layer containing page information and authentication for the admin page.

2.1.2 Memory Constraints

We are subject to the disk space and load balancing of requests to the silver surfer.

2.2 Product Functions

Silver Surfer will provide a more user-friendly interface for both the students and professors in the CS department. It will allow students to look at course syllabuses, contact professors, and find helpful resources. This program will also help professors easily update course information and connect with students.

2.3 Similar Systems

The existing Silver webpages currently exist providing ideas for content for the Silver Surfer project. Inspiration for the layout was taken from systems like Discord, but the functionality is unique.

2.4 User Characteristics

The intended users are as follows: professors in the UMU CS department and CS students. For professors, it is assumed they have a degree and are teaching for the CS department. It is also assumed they are capable of coding. As for students, these fall into two categories: students visiting professor/course webpages on Silver and students running clubs for the CS department. Students who are visiting webpages are expected to have the basic understanding of the web and webpages. Students who are running clubs for the CS department may use the template as

a basis for their webpages or utilize something else. They are expected to have the same understanding as professors in the CS department.

2.5 User Objectives

- Provides an improved user-friendly interface for all clients to use.
- Provides course information for all classes offered in a given semester.
- Provides club information for any active CS department club websites/webpages.
- Provides information on professors in the CS department.

2.6 Constraints

- Silver Server limitations
- Scheduling/Time: Time is not unlimited, making time management and scheduling important.
- VPN Access: If the client is unable to connect directly to Mount Union's network, they will need access via a VPN to access the Silver Server webpages.

3 Specific Requirements

3.1 Functional Requirements

Priority Scale: Low (1) – Medium (2) – High (3)

1. Low: Items that can be eliminated should the need arise, without adversely affecting the product. These items are not urgent and not as important to the final product.
2. Medium: Items that are desired by the customer and/or users of the system, but that may be postponed until a future release. These items are not urgent and but are important parts of the final product.
3. High: Items that are mission critical and without which the system cannot function in a manner that is satisfactory to the customer. These items are urgently needed and important to the success of the final product.

1. **The system has a responsive design**

    1.1. Description
    The design layout should be designed with responsive web in mind. Webpages should respond to window size so that all content is mobile and web friendly.

    2.2. Priority
    3

    3.3. Technical Issues
    The user will need to be connected to the Mount Union network.

    4.4. Risks
    Some risks may include incorrect responsive design making some of the information hard to understand.

    5.5. Dependencies with other requirements
    The consistency of the layout is dependent on this because if the web page is not responsive then the layout will not remain consistent.

2. **The code is well-organized and documented**

    1.1. Description
    The code should be organized logically with descriptive file names and variables to prevent confusion. Comments should be added to explain the code and break it into sections. Spacing and layout of code will be used for easy readability.

    2.2. Priority
    3

3.3. Technical Issues
You will need to have administrative access to see the source code.

4.4. Risks
Some risks would be someone misunderstanding the code when trying to make a change resulting in an error.

5.5. Dependencies with other requirements
Most of the requirements will depend on this because if the code is not well-organized many of the features may contain errors when running the program.

3. The layout is consistent and familiar to use

1.1. Description
The layout will not reinvent the wheel, but rather bring together the best features from other systems and adapt them. The layout will be simple, so it is easy for students to navigate the system.

2.2. Priority
3

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
If the layout is not consistent/familiar it could lose the accessibility aspect of it.

5.5. Dependencies with other requirements
This is dependent on responsive web design and the ability to remain consistent across all pages.

4. There is an admin page that can be used to create pages

1.1. Description
Admins (professors) will have special access to create their own pages on the Silver Server.

2.2. Priority
3

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
Some risks may be revolved around the idea of the admin user needing a password so that only they can make changes to a page or create one.

5.5. Dependencies with other requirements
This will be dependent on the database and the storage of the admin's password.

5. The system contains info from Silver

1.1. Description
The content of the Silver Surfer project will be current content that exists on Silver already. Fake test data is not necessary.

2.2. Priority
3

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
Some risks might include losing connection from the database, thus losing information from Silver Server.

5.5. Dependencies with other requirements
This will be dependent on the database created and storing the information currently being used on Silver Server.

6. The system is accessibility friendly

1.1. Description
The system will be user friendly in the aspects of accessibility which includes picking web friendly fonts and colors that are easy to read.

2.2. Priority
3

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
Some risks might be that it will affect the light/dark mode.

5.5. Dependencies with other requirements
This requirement is dependent on the layout remaining consistent among all pages and being familiar to the users.

7. The system has a database

1.1. Description
Silver Surfer will have a connected database to store commonly used data such as, but not limited to, professor names, professor emails, course names.

2.2. Priority
3

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
If connection to the database fails, Silver Server could lose its output.

5.5. Dependencies with other requirements
This will be dependent with the information from Silver Server being shared on these new web pages.

8. The system can be packaged for production

1.1. Description

Silver Surfer will need to be bundled to run on the silver surfer. This also sets an environment for continuous integration

2.2. Priority
3

3.3. Technical Issues
Blah

4.4. Risks
Blah

5.5. Dependencies with other requirements
Blah

9. The system has a landing page

1.1. Description
The Silver Surfer system will have a landing page with content such as the Mount Union logo. It will have a side navigation bar with course numbers, professors, and an et cetera section. Once a section is selected, a main UI will display with content relating to the selection.

2.2. Priority
2

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
Blah

5.5. Dependencies with other requirements
This will be dependent on the accessibility requirement and the responsive design requirement.

10. There is Markdown Compatibility for content pages

1.1. Description
For professors to have the ability to easily share course syllabuses and any other documents

2.2. Priority
2

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
If the markdown capability does not work the users will not be able to view documents.

5.5. Dependencies with other requirements
This requirement is dependent on organized code and also sharing information from Silver Server.

11. There is a light/dark mode

1.1. Description
The Silver Surfer system will have the option of either a light mode or a dark mode for users.

2.2. Priority
1

3.3. Technical Issues
May not work well on different devices.

4.4. Risks
May conflict with the accessibility requirement.

5.5. Dependencies with other requirements
Ties in with the aesthetically pleasing and accessibility requirements.

12. It is aesthetically pleasing

1.1. **Description**
The Silver Surfer system is aesthetically pleasing to the user's eye.

2.2. Priority
1

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
May conflict with the accessibility requirement.

5.5. Dependencies with other requirements
Ties in with the accessibility requirement, as well as the light/dark mode requirement.

13. There is a chat room for each class

1.1. Description
Each class will have a feature where all students in the class will be able to talk to each other about course work or questions they might have.

2.2. Priority
1

3.3. Technical Issues
The user will need to be connected to the Mount Union network.

4.4. Risks
Some risks may include the inability to prevent students from sending inappropriate messages.

5.5. Dependencies with other requirements
This will tie in with the idea of accessibility providing students a new form of communication with their classmates.

3.2 User Interface Requirements

3.2.1 User Interface: Graphical (GUI)

A graphical interface will be provided for students to navigate between the different course pages, professor pages, and organizations within the department. There will be a landing page that provides access to each of the courses' webpages.

Administrators will also be able to see each of course webpages with the addition of the ability to add or subtract any information.

3.2.2 Application Programming Interface (API)

This project will not include the use of any APIs.

3.2.3 Diagnostics (Error Reporting and Usage Logs)

Errors will be able to be viewed by users with the use of the alert boxes and the use of the 401: page not found error.

3.3 System Requirements

Access to the Silver Server is needed. Everyone should have access to the webpages through the Silver Server. There will also need to be a database to hold course information, administrative personnel info and passwords.

3.3.1 Hardware Interfaces

There are not any designated hardware interfaces that need to be used for this project.

3.3.2 Communications Interfaces

In production, a local port will be exposed for SQL database connections and the Next.js app server.

3.3.3 Software Interfaces

Students will need access to a computer with any web browser and access to the Mount Union network.

Administrators will need access to:

- SQL Server
- Code editor

3.4 Domain Requirements/Constraints

Silver Surfer will only be accessible to clients who are either connected to the Mount Union Wireless Network or the VPN.

3.5 Non-Functional Requirements

3.5.1 Reliability

The Silver Surfer webpages will be a reliable source for course information for any user that is connected to the Mount Union. These pages will also reliably connect to a database to provide stored information to the users. In addition, the administrators will be able to reliably update information and create new pages.

3.5.2 Availability

These pages will be available to all student, faculty, and staff that are either connected to the Mount Union Wireless network or are using the VPN.

3.5.3 Security

Silver Surfer will need authentication to gain access to the admin page to create class pages. There will also need to be authentication for pages to be edited by the professor. The rest of the app is read-only with exception to the class messenger.

3.5.4 Maintainability

To ship code to production, Docker containers will be made to package the app for production.

3.6 Logical Database Requirements

Need to consult Dr. Kirchmeyer for SQL database

4 Software Life Cycle Model

4.1 Choice of Software Life Cycle Model

The Software Life Cycle Model the Silver Foxes has chosen is Kanban.

4.2 Justification for Choice of Model

We chose this SLCM because it seemed the most logical for completing our list of requirements, and it makes the most sense for our current social context compared to other SLCMs such as SCRUM. Our schedules do not allow for daily meetings, so we need a SLCM that does not limit us with meeting requirements.