CoSpace
Software Requirements Specification

Table of Contents

1. Introduction

This section of the *Software Requirements Specification* document (henceforth referred to as *SRS*) provides the purpose of the document, scope of the document, definitions of the terms and abbreviations used in the document, the definition of verbal forms, and an overview of the document's structure.

1.1 Purpose

The purpose of this document is:

- To describe the quality attributes of the system, and the constraints that the design options must satisfy to deliver the business goals, objectives, or capabilities
- To capture functional requirements and their behavioural nature using use case models (*Use Case Definitions* document)
- To capture functional requirements that are not captured in use case specifications in *Use Case Definitions* document
- To discuss between, and select from, competing design options
- To assess the viability of the proposed system
- To understand the service-level requirements for operational management of the solution

1.2 Scope

CoSpace is an online social interest e-club web application designed and developed by *overengineers* team for *BBM384 Teaching Staff* under the scope of Software Engineering Lab. Course Term Project. The application provides a common space for users to engage in conversations about certain topics, form clubs and sub-clubs (a more specific club under a parent club), plan online or onsite events regarding their sub-clubs.

1.3 Definitions

Below are the definitions of the terms and abbreviations used throughout this document.

Term or Abbreviation	Definition
Auditing (audit logs)	An audit log (also called audit trail) is a security-relevant chronological set of records that provide documentary evidence of the sequence of activities that have affected at any time a specific operation, procedure, event, or device.
HTTP/HTTPS	Hypertext Transfer Protocol/Hypertext Transfer Protocol Secure

1.4 Overview

This document, along with the attachments *Use Case Definitions* document, *Graphical User Interface Design* document and *Test Case Definitions* document, defines both the functional requirements, that is, the features and functions of the application, non-functional requirements (also known as "quality requirements"), as well as how to test them, and any other constraints to be considered during and/or after development of the system.

Requirements of the system are detailed with Use Case Driven Development Practice in mind, therefore, most of the functional requirements are modeled using use case definitions in the *Use Case Definitions* document as an attachment to this SRS. Therefore, section 2, *"System-Wide Functional Requirements"*, of this document describes the functional requirements of the system that are system-wide and haven't been described in use cases.

In the third section of the SRS, *"System Qualities"*, non-functional requirements such as performance, stability, usability are described both quantitatively and qualitatively. In both sections 2 and 3, the requirements are given unique identifiers to provide traceability. Traceability is also provided for use cases in *Use Case Definitions* document (For example: *UC12.3* refers to the third path item of the use case UC12).

Fourth section of the SRS, *"System Interfaces"*, describes the requirements to be considered while developing the user interface (not the user interface itself), and the requirements for the interfaces to external systems, both hardware and other software.

Fifth section of the SRS, *"Business Rules"*, describes the requirements of the business domain in which the system shall fit. These include business rules and policies that the system functionality must conform to.

Sixth section of the SRS, *"System Constraints"*, describes any design; implementation or deployment constraints on the system being built that must be adhered to.

Seventh section of the SRS, *"System Compliance"*, describes any usage restriction requirements, applicable standards, and legal requirements the system must be in compliance with.

Eighth section of the SRS, *"System Documentation"*, describes the requirements for user manuals, help systems, and any sort of documentation for future usage of the system.

2. System-Wide Functional Requirements

Most of the functional requirements are captured using use case modeling technique in *Use Case Definitions* document. This section provides the functional requirements that are system-wide, that is, they affect multiple use cases, and/or haven't been represented or cannot be represented in use case descriptions.

As with the *Use Case Definitions* document, the prioritization of the system-wide requirements is made using MoSCoW model[1]. Below are list of terms and their meanings:

- ***Must have:*** Requirements with this priority are essential for the product, and must be implemented.
- ***Should have:*** Requirements with this priority are not essential for the product to work. However, they are nearly as important as the must haves and are therefore expected to be implemented.
- ***Could have:*** Requirements with this priority are a nice addition to the product, and may be implemented, if time and budget allow this.
- ***Won't have:*** Requirements with this priority will not be implemented in this version of the product, but may be nice to implement in future versions.

2.1 Auditing

SWFR1	The system shall keep the IP address used to send a post to the club/sub-club feed by a member.	*Must have*
SWFR2	The system shall keep the IP addresses that have been used by a member.	*Must have*
SWFR3	The system shall keep the records of login and logout activities of a member, along with the following information: Date and time of the activity, username of the member, and the IP address used.	*Could have*
SWFR4	The system shall keep the records of the deletion of clubs/sub-clubs or posts, by administrators (or moderators if applicable) along with the following information: Date and time of the deletion activity, username of the administrator (or administrator if applicable), and the IP address used.	*Could have*

2.2 Authentication

The access control requirements of the system is described in detail for each use case in *Use Case Definitions* document.

2.3 Licensing

SWFR5	The system shall provide a user interface that lists the licenses of the open source software that have been used in any part of the software.	*Must have*

2.4 Reporting

SWFR6	The system shall provide the administrator with the ability to view following usage information of the system in administration panel user interface: Monthly active (that is, users that have logged in at least once in the last month) number of members, active number of visitors in the last hour, monthly number of posts have been sent by the members to the system.	*Could have*

[1] MoSCoW Prioritisation, https://www.coleyconsulting.co.uk/moscow.htm

2.5 Scheduling

SWFR7	The system shall send a deletion notice to the members of a sub-club after that sub-club has not been active (meaning no content has been posted) for 90 days.	*Must have*
SWFR8	The system shall delete a sub-club if it has not been active for 7 days after a deletion notice is sent to the members of that sub-club.	*Must have*

2.6 Security

SWFR9	The system shall convert the plaintext passwords of the users into a *hashed* format during registration using an industry standard hashing algorithm.	*Should have*
SWFR10	The system shall use a secret *salt* during the hashing procedure of the users plaintext passwords.	*Could have*
SWFR11	The system shall encrypt the private messages between users using an industry standard encryption algorithm, and store encrypted versions rather than plaintext versions in the database.	*Could have*

3. System Qualities

This section specifies the required system quality factors that are different than the specific functional requirements described in *Use Case Definitions* document and Section 2 of this SRS.

3.1 Usability

SQR-U1 The system shall provide an easy-to-learn interface, that is, at least ⅘ of the users shall not spend more than 5 minutes to learn to interact with the system.

SQR-U2 The system shall provide an easy-to-reach design, that is, navigation within the system should be obvious and the structure (tree) of the website should not be deep.

SQR-U3 The system shall provide an easy-to-remember page layout.

SQR-U4 The system shall provide easy-to-understand system messages.

SQR-U5 The system shall ensure here are satisfied users.

3.2 Reliability

SQR-R1 The system shall provide a response with high accuracy when a request come and all calculations shall provide correct result.

SQR-R2 The system shall provide at least 99% uptime for availability.

SQR-R3 The system shall provide recoverability with the support of cloud system.

3.3 Performance

SQR-P1 The system shall provide a response time of maximum 400ms for search functionality, excluding the internet response time of the user.

SQR-P2 The system shall provide a response time of maximum 250ms for real-time messaging functionality, excluding the internet response time of the user.

SQR-P3 The system shall provide a response time of maximum 250ms to populate the opened club/sub-club feed, excluding the internet response time of the user.

SQR-P4 The memory usage of the system shall not exceed 500MB at any time.

3.4 Supportability

SQR-S1 The system shall work without any installation on Chromium based browsers or Firefox browser.

SQR-S2 The system shall provide a contact form to support the users.

SQR-S3 The system shall scale easily with support of cloud system without any interruption.

4. System Interfaces

4.1 User Interfaces

4.1.1 Look & Feel

The system theme should be green and dark-blue weighted. These colors are similar with overengineers's theme colors. It should be a monolithic design. Considering the popularity of the dark theme, these colors will be attractive for users. Material Design[2] philosophy should be followed.

4.1.2 Layout and Navigation Requirements

There should be 5 main sections in the layout of the system. On the top, there should be header which includes menu links, search box, and login components. On the left, there should be a club list. On the right, there should be action components such as club info, events info, and new club component. In the middle of the page, there should be a feed which contains appropriate posts with their title, body, time, and author information. Also, there should be a vote component on the left of post component. Finally, there should be a footer at the bottom of the page.

4.1.3 Consistency

The system should have responsive design for to ensure consistency. Some navigation components should minimize and enlarge with a button instead of all view to satisfy responsive design constraints.

4.1.4 User Personalization & Customization Requirements

There should be some authorized content in the system such as admin page, moderation page, and etc. Also, members should not have permission to view clubs they have not registered before.

4.2 Interfaces to External Systems or Devices

4.2.1 Software Interfaces

There should be interface for Zoom or When2Meet to show information about the events to members of any club.

4.2.2 Hardware Interfaces

There should no need for any hardware interfaces to use the system.

4.2.3 Communications Interfaces

The server of the system and the browser of the member should communicate with HTTP/HTTPS protocol via infrastructure of Java Spring framework.

[2] https://material.io/design

5. Business Rules

5.1 Registration & Login

5.1.1 BR1 - Requirement for Registering in the System

5.1.1.1 Users under the age of 13 cannot register on the system.
5.1.1.2 Admins can set a special age limit to enter specific clubs/sub-clubs

5.2 Code of Conduct

5.2.1 BR2 - Uses of Slang Word

5.2.1.1 If the member of the system uses slang words in public/private messages, or during the online/offline event or be complaint about him/her, the sub-club admin should ban this member.
5.2.1.2 The banned member cannot join this sub-club for the following 5 days and cannot join the following online and offline events.
5.2.1.3 If the member is banned more than 3 times, the sub-club admin can dismiss him/her from the sub-club.

5.3 Privacy

5.3.1 BR3 - Information CoSpace Collects

5.3.1.1 The system collects information to provide better services to all CoSpace's users. When a member signs in, the system collects information that members provide like e-mail addresses, name, surname, age and password. These are treated as personal information.
5.3.1.2 The system may collect users' activity information: Terms are searched for, people with whom users communicate, etc.

5.3.2 BR4 - Sharing User Information

5.3.3 We do not share any user information with any other customer or services.

5.3.4 BR5 - Deleting User Information

5.3.4.1 A member of the system must be able to delete their account at any time.

6. System Constraints

The system is a cross-platform web application. Spring framework for backend server, and ReactJS for frontend system will be used. The system will use SQL databases to manage data.

7. System Compliance

7.1 Licensing Requirements

The system shall be released under the Privity of the BBM384 Software Engineering Lab Course of Hacettepe University. All the source codes will be private until the course completion. After that, all the source code will be publicly available on our GitHub page.

7.2 Legal, Copyright, and Other Notices

By submitting user content to CoSpace, the user grants us a royalty-free, perpetual, irrevocable, non-exclusive, unrestricted, worldwide license to reproduce, prepare derivative works, distribute copies, perform, or publicly display the user's content in any medium and for any purpose, including commercial purposes, and to authorize others to do so.

7.3 Applicable Standards

The system should comply with the personal data protection laws such as GDPR and KVKK and copyright laws in the countries where it operates.

8. System Documentation

The system should provide a user interface for the documentation regarding the functions of the application. It

should also provide a contact form as described in quality requirement **SQR-S2**.