Software Requirements Specification

for

ANALYTICA

Version 1.0

In partial fulfillment of the requirements of CMSC 128

CMSC 128 D-4L

May 13, 2021

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Marty de Viterbo	May 13, 2021	Initial version	1.0

1. Introduction

1.1 Purpose

The purpose of this document is to provide features and specifications of Analytica UPLB-ICS Online Library Version 1.0 which is for the use of the Institute of Compute Science. This document contains the functional and non-functional requirements of the project to serve for client and software developers. Due to the pandemic and limited face-to-face interaction, this project provides an online library based on the physical library's availability of resources such as books, journals, special problems, and theses to give and provide an online material access to students, staffs, faculty, and administrators.

1.2 Document Conventions

The title page of this document is written in Arial font with sizes 32 and 20 for the titles, while 14 for the subtitles. Headers are written in bold Times New Roman, with section headers at size 18 and subheaders at size 14. Paragraphs are all aligned as justified. The rest of the text in this document is in Arial, size 11.

1.3 Intended Audience and Reading Suggestions

This document is intended for the project developers (project manager, team leaders, members, and the instructors) and users (guests, students, ICS administrators, ICS faculty, ICS staff). It is recommended for the readers to at least scan and skim the document from top to bottom to get an overview of the basics. For readers with extensive knowledge of the system can skip through the document, as desired.

1.4 Project Scope

The project aims to give a solution by providing an online access to guests, students, faculty, administrators, and staff by creating a corresponding online library from the Institute's physical library. It helps the student and faculty to read and cite electronic resources for their current or future studies that are within their special interests, as well as to manage and keep the resources inline to the university and institute's policy. Features may be added to the system in the future, to make it general-purpose, and inline to the interest of providing an additional information and making processes more convenient.

1.5 References

This document is based on a template downloaded at. Copyright to Karl E. Wiegers, but permission is given to access, edit, and copy the material, as stated in the title page of this document.

2. Overall Description

2.1 Product Perspective

The system which provides an online library to cite or read by guests, students, staff, and faculty as well as to manage by administrators, comes from the idea of ICS faculty members: Asst. Prof. Reginald Recario, Asst. Prof. Fermin Roberto Lapitan, Asst. Prof. Johaira Mae Arriola, and Asst. Prof. Kendall Jeane Jaen. This project is assigned to CMSC 128 D-4L by Library Head Committee Ariel Doria. Features outside the scope of the project may be added to expand the system's features and specification.

2.2 Product Features

The system will:
- *Allow user authentication*
- *Allow user to update and delete his/her account*
- *Allow user to browse, search, and/or use filter queries on SP, theses, books, and authors*
- *Allow guest to only see the list and abstract of SP, theses, and books*
- *Allow student to only see the list, abstract, poster, and journal of SP and theses, also the list, abstract, and content of the books*
- *Allow administrator to add, update, view full access, and delete SP, theses, and books*
- *Allow administrator to manage users*
- *Allow administrator to view activity logs of users*
- *Allow administrator to view and/or export summary report of SP, theses, and books*
- *Allow user to see the latest acquisition of books*
- *Allow user to see the latest UPLB News*

2.3 User Classes and Characteristics

Users are classified into five: administrator, staff, faculty, student, and guest. Administrator, staff, faculty, and student are all constituent of university, or by using organizational email. Guest is considered as guest if the user is not logged in to the system and is to be considered as not part of within the university.
Administrators have full access to the system features and specifications. This includes full view access of all resources (SP, thesis, and book), together with create, update, and delete of both resources and users. In addition, the administrators have the only access of viewing and/or exporting summary reports same as with activity user logs. Faculty and staff have the full access for viewing resources but have no privilege to do any modification such as add, update, or delete for both resources and users. Student has access to SP and thesis's list, abstract, journal, and poster only while has full access to the book particularly to book's list, abstract, and its content. Guest can access SP, thesis, and book's list and abstract part only (see Appendix B Use Case Diagram).

2.4 Operating Environment

The system uses MERN (MongoDB, Express, React, Node) technology stack to operate easier and faster web application. The system back-end is stand-alone and can run in any operating system with NodeJs installed. The database can be viewed by either using a browser or by installing MongoDB Compass on local machine. The front-end is also stand-alone and will require client environment to have a browser capable of handling HTML, CSS, and Javascript.

2.5 Design and Implementation Constraints

The development of the system's implementation is limited by the following: time as it is an academic requirement for CMSC 128, and mode of communication as it builds concept barrier among the developers during implementation caused by the pandemic. Limited resources such as unstable internet connection affect the availability of each member of the team which influence the productivity of the team, thus the project timeline. The team will not be responsible for system deployment as well as for the maintenance.

2.6 User Documentation

There is currently no user manual available for the system.

2.7 Assumptions and Dependencies

The system will be web-based application, and may be deployed through the ICS servers soon enough after the pandemic. ICS servers run on UNIX systems (Linux) where it is assumed it can handle Javascript. The system can be run and accessed locally, and can become more readily available online, thus, stable Internet connection capability is needed. The system feature of getting UPLB recent news may get affected since data or content that is being referenced or web scrapped from the UPLB official website may also change if there is any in the future. Frameworks that are used may be deprecated or become obsolete few years from now after the system is developed, resulting to unusual or odd response compared to what a user is expected to be.

3. System Features

3.1 User Authentication

3.1.1 Description and Priority

User authentication feature is considered as high priority. It is responsible for user login and logout functionalities.

3.1.2 Stimulus/Response Sequences

- *The user clicks the login button.*
- *The user enters his/her valid organizational email*
- *The user is redirected to the home page if successful.*
- *The user clicks his/her name on the dropdown menu.*
- *The user clicks the logout button.*

3.1.3 Functional Requirements

REQ-1: The system must validate user's organizational email
REQ-2: A message must appear notifying if the user either successful or fail to login

3.2 Browse and Search Resources

3.2.1 Description and Priority

Browsing and searching resources is considered as high priority. The feature serves to browse or search a particular resource based on different filters applied such as title, type, and/or author.

3.2.2 Stimulus/Response Sequences

- *The user enters search query string and may select any filter(s).*
- *The user is redirected to the browse page by clicking or pressing enter button*
- *The user may click specific resource to view more details*
- *The user may still modify and continue browsing and searching.*

3.2.3 Functional Requirements

REQ-1: The system must validate search string

REQ-2: Proper webpage must appear if the user successfully input query

REQ-3: A message prompt must appear if the search query input is invalid

3.3 Manage Resources

3.3.1 Description and Priority

Managing SP, theses, and books is considered as high priority. The feature allows the user with the type of administrator to take responsible on managing resources by adding, reading, updating, or deleting resources.

3.3.2 Stimulus/Response Sequences

- *The user clicks the manage resources button.*
- *The list of resources will be displayed with options.*
- *Choosing any on the options are for adding, updating, and deleting specific resource.*
- *The user will click confirm for any changes made*

3.3.3 Functional Requirements

REQ-1: The system must validate the user type of the current user.

REQ-2: The system must display the updated list of resources to the user.

REQ-3: A message prompt must appear if the action is failed or successfully done.

3.4 Summary Report

3.4.1 Description and Priority

Displaying summary report is high priority feature. The feature allows to summarize all the resources in one table, and can be able to download or export to a file. Only the user with the type of administrator can use this feature.

3.4.2 Stimulus/Response Sequences

- *The user clicks the view summary report button.*
- *The list of resources will be displayed with options.*
- *The user clicks the download button.*
- *A file having the content of summary report will be downloaded.*

3.4.3 Functional Requirements

REQ-1: The system must validate the user type of the current user.

REQ-2: The system must display the updated list of resources to the user.

REQ-3: A message prompt must appear if the action is failed or successfully done.

3.5 Activity Logs

3.5.1 Description and Priority

Displaying activity logs feature is considered as low priority. The feature allows the user with the type of administrator to take responsible on observing and seeing the user's activity logs which include user's date of logging in and logging out of the system.

3.5.2   Stimulus/Response Sequences
- *The user clicks the activity logs button.*
- *The table of user activity logs will be displayed.*
- *The user clicks the delete button.*
- *The table will be cleared.*
- *The user clicks the download button.*
- *The table of activity logs will be downloaded*

3.5.3   Functional Requirements

REQ-1: The system must validate the user type of the current user.

REQ-2: The system must display the updated list of activity logs.

3.6 Recent UPLB News

3.6.1   Description and Priority

Viewing recent UPLB news is considered as low priority. The feature allows any type of user to view the recent UPLB related news synchronized with the UPLB official website.

3.6.2   Stimulus/Response Sequences
- *The user may view the recent UPLB news' title and cover photo*
- *The user may click the card news but will be opened and viewed externally*

3.6.3   Functional Requirements

REQ-1: The system must be synchronized with the UPLB recent news website

4. External Interface Requirements

4.1 User Interfaces

The user interface is aesthetic-minimalistic or is presented in black-blue-white theme colors approach (see Appendix B User Interface). It will use a pure Model-Controller-View JavaScript architectural pattern that MERN stack uses.

4.2 Hardware Interfaces

Client computers can access the system by using a standard browser that can handle JavaScript and can connect to network. The communication between server and client is done in JSON format.

4.3 Software Interfaces

The database management system used in this project is MongoDB. The system uses Mongoose as object data modeling library to control and manage relationships among data, provides schema validation, and is used to translate between objects and the representation of those objects from Node to MongoDB, and vice-versa. The server, Node used Express web framework for allowing to setup middlewares to respond to HTTP requests. The Node-Express as backend communicates with React via JSON by using Axios which helps the client to make HTTP requests to external resource such as Node server, and vice-versa. The front end and back end components are both stand-alone, which means they can interact with other systems.

4.4 Communications Interfaces

The standard HTTP protocol will be used. User's information is encrypted while user's personal credentials are not stored in the database such as password since the user's main source of information depends only on the response object from Google's API e-mail system which uses an organizational mail to login to.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The system can be accessed both in the local network and through the Internet. Any changes, updates, or modification made by the user is shown in real time with correct response of the data made. The system will respond to the user in at most half to one second from the time of the request submittal. The system will be allowed to take more time when doing huge processing jobs such as mounting at the start of the system.

5.2 Safety Requirements

No safety requirements are specified.

5.3 Security Requirements

Google API is used to authenticate user's organizational email. No personal or private credentials such as Google e-mail's passwords is read or stored by the system. Information of the user is kept encrypted using JWT when passed back to the client side such as for the use of the token stored in the user's browser.

5.4 Software Quality Attributes

The most important attributes for this system are the maintainability, reliability, robustness, and reusability. Maintainability, source codes both in client and server sides are both well-documented so future developers may easily understand as well as to continuously develop the system. Robustness, abnormal and erroneous inputs are handled responsively with error-message saying so. Reliability, the system is available 99.99% of the time. Reusability, designs and implementations can be reused.
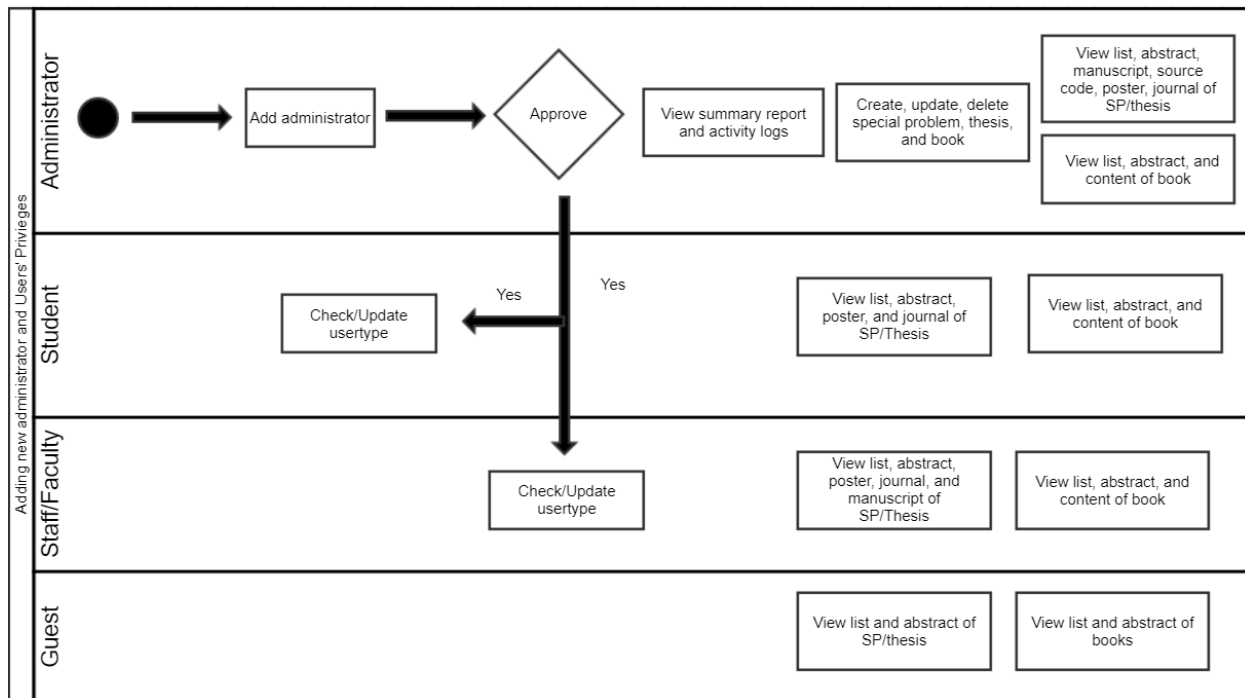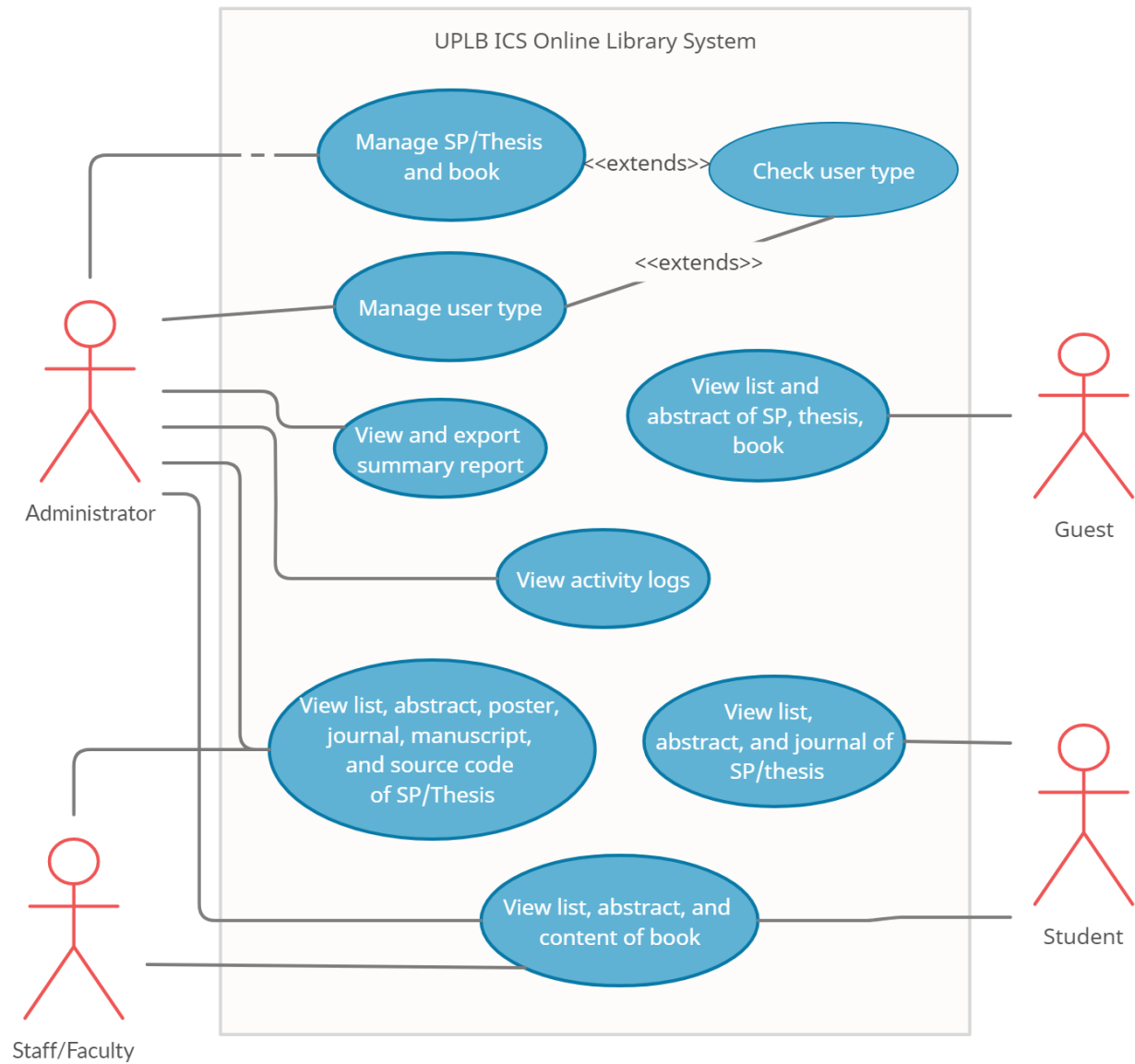
6. Other Requirements

Appendix A: Glossary

- *Admin – administrator of the system*
- *CMSC – computer science*
- *ICS – Institute of Computer Science*
- *Guest – user that is not logged in on the system*
- *HTTP – hypertext transfer protocol*
- *JWT – JSON web token*
- *MERN – MongoDB-Express-React-Node technology stack*
- *SP – Special Problem*
- *Student – a BS Computer Science student of the University of the Philippines Los Baños*

Appendix B: Analysis Models

Activity Diagram

Use Case Diagram

UPLB ICS Online Library System

Manage SP/Thesis and book

<<extends>>

Check user type

<<extends>>

Manage user type

View list and abstract of SP, thesis, book

View and export summary report

View activity logs

View list, abstract, poster, journal, manuscript, and source code of SP/Thesis

View list, abstract, and journal of SP/thesis

View list, abstract, and content of book

Administrator

Guest

Student

Staff/Faculty

User Interface

Appendix C: Issues List

Real time response if there is any updates made in a multiple sessions system.