Study Plan System

Software Requirements Specification

Web and Mobile Application

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 01/01/23 | 1.0 | Initial documentation for the SPS project | Bert Bertington |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Scope 4

1.2 Intended Audience 4

1.3 Definitions, Acronyms, and Abbreviations 4

1.3.1 Student 4

1.3.2 Lecturer 4

1.3.3 Study Plan 4

1.3.4 API 4

1.4 Overview 4

2. Overall Description 4

2.1 Use-Case Model 5

2.2 Product Position 5

2.3 Assumptions and Dependencies 6

2.3.1 Assumptions 6

3. Specific Requirements 6

3.1 Use-Case Specifications 6

3.2 Functional Requirements 6

3.3 Supplementary Requirements 6

4. Supporting Information 7

4.1 User interface prototypes 7

# Introduction

The purpose of this document is to define requirements of the Study Plan System. This Software Requirements Specification lists the non-functional requirements as well as the use cases presented in the system.

## Scope

This specification applies to the Study Plan System.

This specification defines the non-functional requirements of the system, such as reliability, usability, performance, and supportability, as well as functional requirements that are common across several use cases. The functional requirements are defined in the Use Case Specifications.

## Intended Audience

This software requirements specification document is intended for all project stakeholders to use as a guide as to the direction and scope of the Study Plan System.

## Definitions, Acronyms, and Abbreviations

### Student

A user who is enrolled in a TAFE SA course.

### Lecturer

A user who Teaches or is Responsible for Subjects and Students.

### Study Plan

A set of Subjects that are required to complete a qualification.

### API

Application Programming Interface

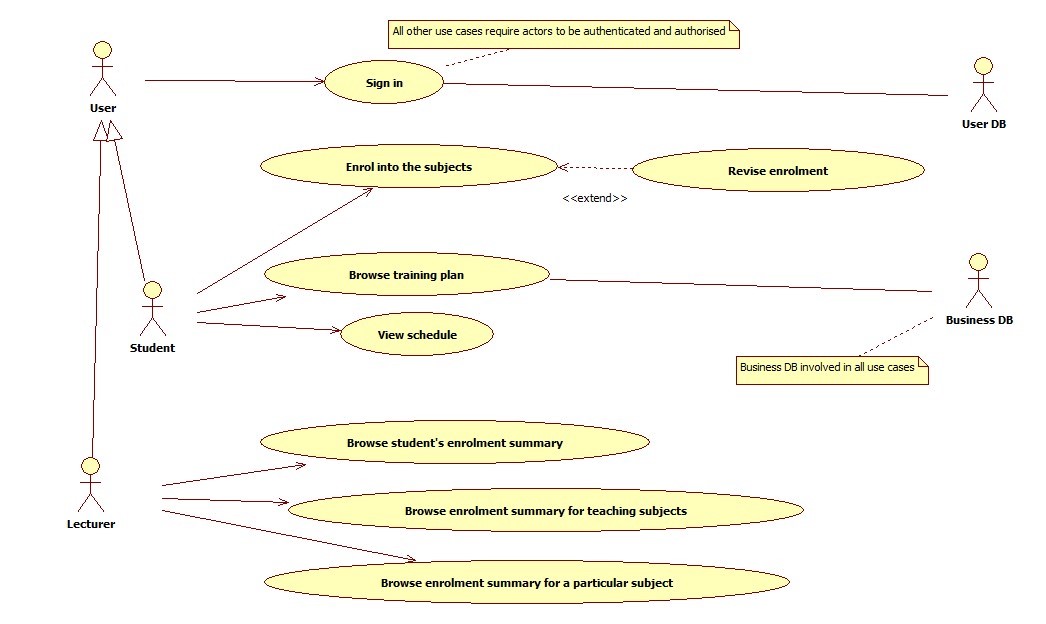
## Overview

The document will list specific requirements, assumptions, and dependencies for this system. This document and use-case specifications together capture the complete set of requirements on the system.

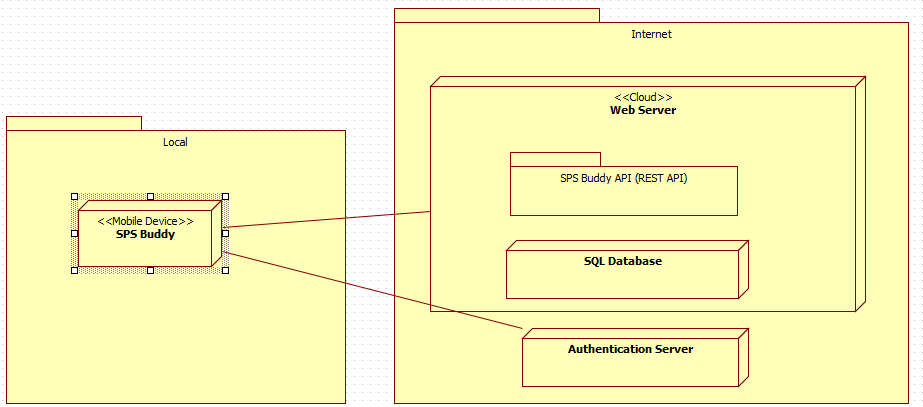
# Overall Description

Students should be able to register for classes for the qualification they are enrolled in. The application should handle the fact that a student can have multiple study plans. A selection of completed and remaining subjects of all required competencies should be automatically updated and shown when a student views their subjects.

## Use-Case Model



## Product Position



## Assumptions and Dependencies

### Assumptions

* The student has a Study Plan
* The student has paid the course fees.
* The student/ lectures have already been registered to the system/database.

The system will depend on the API to access the database. Therefore, the system is highly dependent on the API for working.

# Specific Requirements

* The system will require a Mobile or Web Device
* The system will need access to the internet
* REST API utilised for the back-end data retrieval
* JavaScript framework implemented for faster development
* Must be a web application
* Modern user interface with drag and drop capabilities
* User must be able to log in using their existing lecturer or student details
* View a timetable for subjects on the enrolment screen
* User must be able to see qualification completion percentage

## Use-Case Specifications

The use-case specifications are in separate documents. The following are the use-cases.

* View Study Plan
* Browse Students
* Maintain Subjects

## Functional Requirements

User story: As a lecturer, I need to be able to create a student study plan for a new student.

Functional Requirements:

1. TBD
2. TBD

User Story: As a lecturer, I need to be able to create a student study plan for a continuing student

Functional Requirements:

1. TBD
2. TBD

## Supplementary Requirements

1. *Security:*

* The application ensures that the users perform only the operations for which they have been authorized. Students are not able to view another students’ information.
* Database access will require a username and password.

1. *Reliability:*

* When the database updates, students study plan and enrollment view will update as well.
* The check list will check completed subjects and display them appropriately.

1. *Availability:*

* The application will be available 24/7 if the user is connected to internet unless going through maintenance. Users will be notified through email and will not take more than 24 hours.

1. *Usability:*

* The application will have an intuitive UX. Focusing on keeping a clearly defined UI (i.e., Buttons will be clearly labelled. Navigating through pages will be clear.)
* Users with little understanding of English will be able to use the application.

# Supporting Information

## User interface prototypes

Dashboard Desktop Prototype 1

Graphical user interface, application

Description automatically generated

Login Screen Mobile Prototype 1

Graphical user interface, application

Description automatically generated