Software Requirement Specification

Student Enrolment System

by

██████████████████████████

████████████

Supervisor:
Dr Patrick Then

Team Members:

████████████████████████

██████████████████████████

█████████████

██████████████

███████████

Table of Contents

1.0 Introduction

1.1 Purpose

The purpose of this document is to provide the requirements of the Student Enrolment System. It will specify both the functional and nonfunctional requirements required by the client in order to develop a fully working Student Enrolment System.

1.2 Intended Audience and Reading Suggestion

This document is intended for:
- **Developers:** Delivers the end product for the client while fulfilling the requirements specified in this document.
- **Clients:** To check on the requirements specified in this document.
- **End Users:** To give them an idea of what will be expected from the Student Enrolment System.

1.3 Project Scope

New Features

The scope of this project is an improvement of the existing Student Enrolment System in ██████████████. The system will provide features for **adding and dropping units, viewing the Unit Listing** after for continuing students. Furthermore, **viewing Course Study Planner**, applying for an **Internal Course Transfer** and **Contacting Course Coordinator** will be implemented for students. Course Coordinators will be given the ability to **add, modify or remove units and their prerequisites**. Units added to the system can then be **added and removed from the Unit Listing and Study Planner**. Course Coordinators will also be able to **resolve student enrolment issues** that may come up during enrolment such as enrolling beyond the units recommended by the Course Coordinator. Student Administrators will be able to **Open and Close Enrolment** as well as **adding, modifying and removing students** from the enrolment system. Both the Course Coordinator and Student Administrator will be able to **import in bulk** a list of units and students respectively.

Unimplemented Features (Future Features)

Several features will not be implemented in the first iteration of the Student Enrolment System. These features are in consideration and will be implemented in future iterations of the project. Among these are **Application for Leave of Absence**, **Application for Exemptions**, **Deferring an Offer Letter**, **Application for Withdrawal from Program**, **Application for Irreconcilable Timetable Clashes** and the option for **Student Administrators and Course Coordinator to modify business rules** in the system.

1.4 Definitions, Acronyms and Abbreviations

Below are definitions of terms used within the domain of the ▆▆▆▆▆▆ Enrolment System. Definitions are further are covered in the next section (1.5 Business Rules).

Terminology	Definition
Dependencies	The software needed so the system will work without problems
Responsive	A design technique to ensure the software scales through different screen dimension and resolution
Virtual Server	A server (computer that hosts the system) built in a virtual machine
Unit	A subject of study that students can enrol into
Course	A study path students can register into
Study Planner	A list of units for a specified intake to plan their enrolments throughout their study
Unit Listing	A list of units that will be available in the next enrolment

1.5 Business Rules

This section will provide all business rules that applies to the enrolment system:

Rule	Condition
Maximum Units (per Semester)	Every semester has a an amount specified by the students can take four units (Full load). But, some students are in final semester then they can take maximum five (5) units (Over load).
Invalid Payment Status	Enrolment might be void or rejected because of student's previous payment status.
Withdrawal	Students can withdraw units by 4th week from semester started. If, any student wants to withdraw from any unit after 4th week then the fee will be forfeited.
Internal Course Transfer	Sometimes some students want to change their degree. Student can apply to change or transfer to other Degree by only the approval through enrolment system.
Final Year Project	It is compulsory for students to take FYP part B or 2 in the next semester, if students take FYP part A or 1 in the first semester.
Special Consideration	An exceptional case where a student is taking more units than the limit specified
Leave of Absence	A condition that allows students to put their studies on hold for a certain amount of time
Exemption	A condition where students will not be required to take a specific unit of study
Offer Letter Deferment	A student may choose to forfeit their offer letter before their studies commences. International students are not allowed to defer an offer.
Withdrawal from Program	A student may choose to forfeit their enrolment to a program in █████████
Prerequisite	A unit that is required to be completed before enrolling the specified unit.

2.0 Overall Description

2.1 System Perspective

Student enrolment System is a system that allows both international students and local students to enrol into subjects of their respective courses. This system incorporates business rules which covers a wide range of regulations and policies such as *pre-requisites, payment status, course coordinator decisions* and more. Not only that, this system also incorporates notification system where the students will get notified once their enrolment has been accepted or rejected.

2.2 New System Features

There are three main actors within the system. Therefore, the features of the system is categorized based on the actors. Below are the features that will be implemented in the first iteration:

Students
1. Add/Drop Subject Units
2. View Unit Listing
3. View Study Planner
4. Internal Course Transfer Application
5. Contact Course Coordinator

Student Administrator
1. Set Enrolment Dates (Activate/Deactivate Enrolment)
2. Add/Modify/Remove Student enrolment and details

Coordinators
1. Add/Modify/Remove Units
2. Add/Remove/Edit Units in Study Planner
3. Add/Remove Units in Unit Listing
4. Resolve Student Enrolment Issues

2.3 User Classes and Characteristics

Client (Mr ███████████████)
The client is the product owner, who proposes to have the system implemented. As a lecturer in ██ he felt that there is a need to improve the current system that relies on a manual process to make it more accessible and user-friendly to all users.

Students
The students are the primary user of the system. During the Final Exam period (nearing the end of the semester), students will apply for re-enrolment into subject units for the next semester. However, the students will still need to check for their enrolment status in Phase 2; whether it is approved or rejected.

Student Administrators
Currently, the student administrator are the ones who processes the enrolment for each student. They are the ones who process the business rules (apart from prerequisites). They are also responsible to open the enrolment and close it at the designated times.

Coordinators
Different courses have different coordinators to manage the students and subjects in their respective courses. Their purpose in the system is to decide the prerequisites and the offered subject units in the next semester and give approval for matters. As the course manager, he can also directly interfere with the system should a *special consideration* arise.

2.4 Design and Implementation Constraints

Language
English will be the primary language used in the Student Enrolment System. English was selected as ██

The User Interface Design
The user interface is required to scale across different screen resolutions (responsive design). The design will be focused towards being mobile-friendly, while smoothly transitioning to higher resolution devices.

Security
The system must be secure and users will only be able to access the system to the extent allowed by their access level.

Relevance of Other Existing Systems
The system will be required to integrate with existing systems that are currently used in ▬▬▬▬▬▬

2.5 Assumptions

1. All users can read and understand English language.
2. All users have basic knowledge in using a computer.
3. All users know how to navigate through a simple management system.
4. The network data transfer rate will be a minimum of 2.5Mb/s.

2.6 Operating Environment

As for the requirement of the server, we will host it on Digital Ocean, a company that provides **virtual server hosting(cloud computing)**. The minimum specifications are as follows:

| Memory/CPU	512MB / 1 CPU
Storage	20GB SSD Disk
Transfer	1000 GB

Operating System	Fedora 23 Server
Web Server	Nginx 1.9.3
Backend	PHP 5.6, Composer, Npm, Laravel 5.2
Frontend	Javascript, HTML5, CSS3
Database	MariaDB

Currently, with the minimum specification, the server is expected to able to accommodate the whole system. Currently, ▬▬▬▬▬ students numbers up to 4,000 to 5,000 people. Storing data would be no problem but when this number of people accessing the system on the same day, there might be a drop in performance. This would then still be okay based on past experience, however, in 5 to 10 years in the future when the number of students doubles up to 10,000 people and accessing the system at the same day, there will be a drop in performance and potentially detected as a DDoS attack. Thus, a stronger droplet *4GB RAM/2 CPU* may be required and this time. A server specification upgrade to improve performance can easily be done as it is hosted on a virtual server.

3.0 System Features

3.1 Current Practice(Old System)

In the current practice, the Student Administrator will announce the date and duration of the re-enrolment which opens in the latter half of the semester. Students will choose to enrol the subject units of their choice. The student admin will then collect information from the online form and manually key-in the student enrolment into the current system. Two to three weeks before the next semester commences, the students will receive an email regarding their enrolment details together with an invoice. The student will then be given time until one week after the semester commences to pay the tuition fee and only then will their enrolment be confirmed and thus become eligible to sit for the Final Examination.
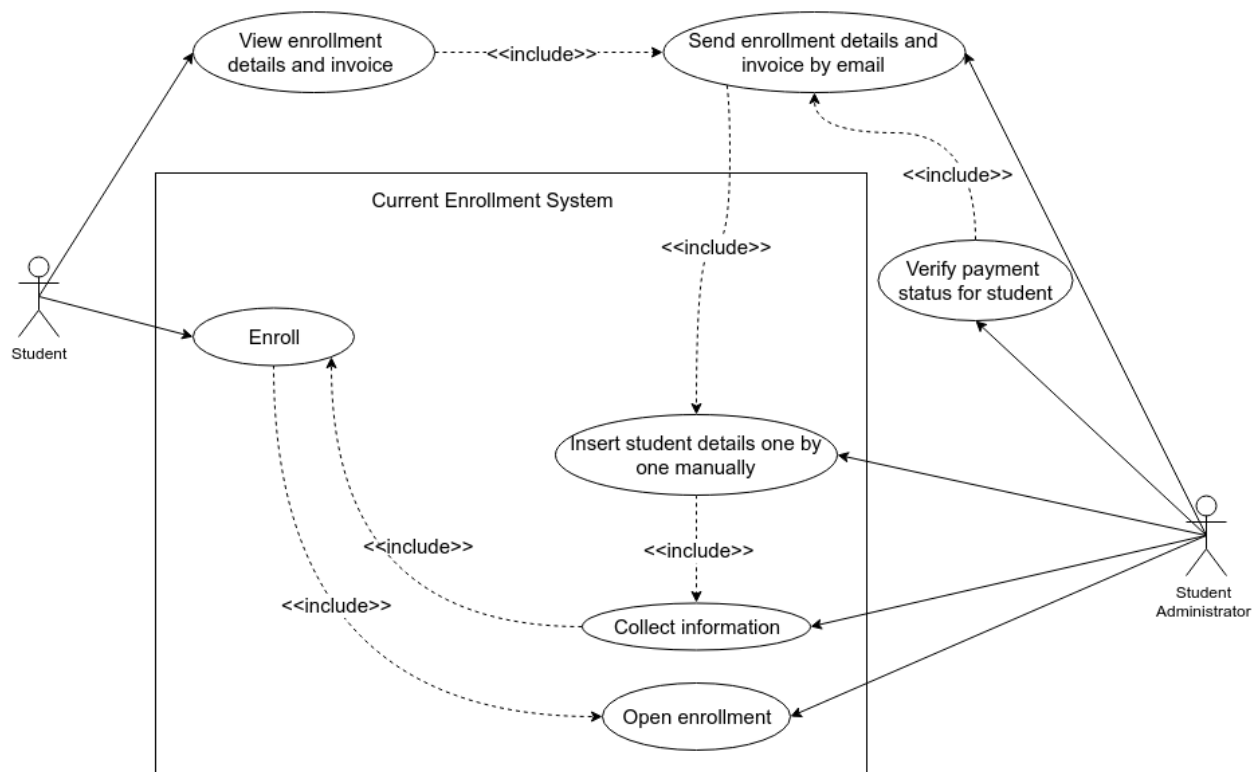
3.2 Problem with Current Practice

- Students may not notice updates to units offered in the next semester(e.g. removal of unit)
- Business rules are manually checked by administrators
- The workflow is tedious because the system is not centralized

● The online form is not mobile-friendly

3.3 New System Features/Functionalities (Use Cases)
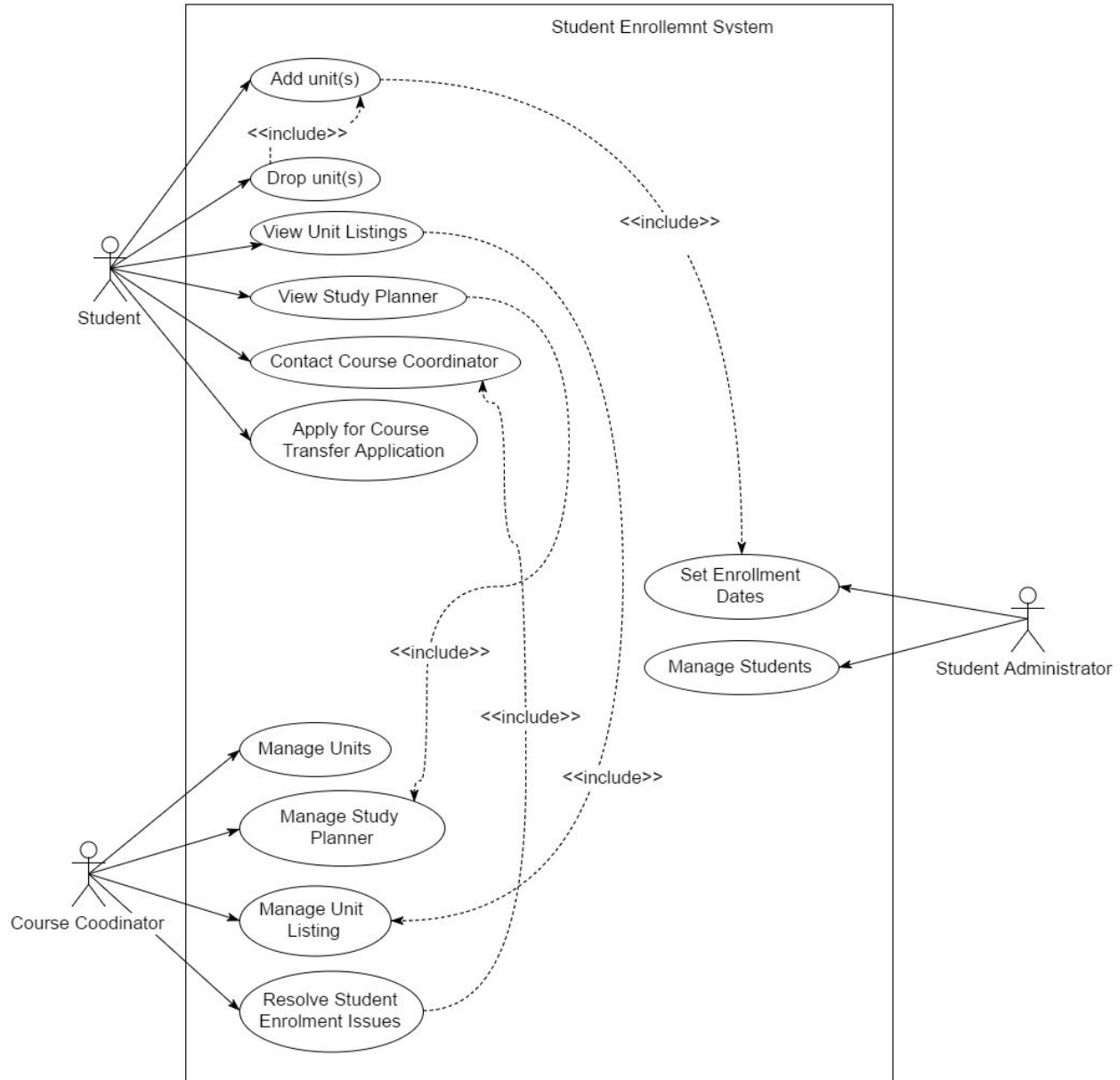
The system will assist the enrolment process by checking business rules as specified by the Student Administrator and Course Coordinator. Enrolment information will be easily accessible to all users interacting with the system. Below are the tasks each user can perform.

Actor: Students

Task 1	Add Subject Unit(s)
Purpose	To add one or more unit(s) during amendment
Precondition	-
Success Guarantee	One or more units are added to enrolment and a notification will be sent to the student's email

Sub-Tasks / Scenario
1. Student logs in to the system
2. Student chooses to add units for the current semester
3. System will display a list of the units offered in the next semester
4. Student will choose which unit(s) to be added from the list and confirm the changes
5. System will display a success message to the student and a notification will be sent to the student's email
6. Student logs out of the system

Variant
4a. The semester has already begun
4a1. System will notify the student that the action taken requires a Course Coordinator's approval
4a2. Student confirms action, a notification is sent to the student's email, a notification is sent to the Course Coordinator and the action is logged

Task 2	Drop Subject Unit(s)
Purpose	To drop unwanted units in the previous enrolment
Precondition	Student has at least one unit enrolled
Success Guarantee	The dropped unit(s) will be removed from the previous enrolment and a notification will be sent to the student's email

Sub-Tasks / Scenario
1. Student logs in to the system
2. Student chooses to drop units from the previous enrolment
3. System will display a list of the units chosen in the current enrolment
4. Student will choose which unit(s) to be taken out from the list and confirm the changes
5. System will display a success message to the student and a notification will be sent to the student's email
6. Student logs out of the system

Variant
3a. Student does not have any units enrolled
3a1. System will display a message notifying that there are no units to drop
4a. The semester has already begun
4a1. System will notify the student that the action taken requires a Course Coordinator's approval
4a2. Student confirms action, a notification is sent to the student's email, a notification is sent to the Course Coordinator and the action is logged

Task 3	View Unit Listings
Purpose	To list out the units being offered in the upcoming semesters
Precondition	-
Success Guarantee	A list of units offered in the upcoming semesters are displayed
Sub-Tasks / Scenario	
1. Student logs in to the system	
2. Student chooses to view unit listing	
3. System will display a list of the units offered in the upcoming semester	
4. Student logs out of the system	
Variant	
3a. Student changes the semester the list is displaying	
3a1. Student selects a different semester unit listing to view	
3a2. System updates the list of units displayed	

Task 4	View Study Planner
Purpose	To access the study planner according to the intake
Precondition	-
Success Guarantee	The study planner is displayed
Sub-Tasks / Scenario	
1. Student chooses to view the study planner (default view to their own intake)	
2. System will display the default study planner	
3. Student logs out of the system	
Variant	
2a. Student changes the study planner intake	

2a1. Student choose to view study planner for different intake
2a2. System will display the study planner based on the chosen intake

Task 5	Contact Course Coordinator
Purpose	To allow students to contact the Course Coordinator
Precondition	-
Success Guarantee	An email is generated and sent to the selected Course Coordinator
Sub-Tasks / Scenario	
1. Student logs in to the system	
2. Student chooses the option to contact the course coordinator	
3. System will display a form for the student to fill	
4. Student fills the form with his inquiry and confirms message	
5. System displays a success message, generates an email and sends it to the selected Coordinator	
6. Student logs out of the system	
Variant	

Task 6	Internal Course Transfer Application
Purpose	To allow the student to switch course of study
Precondition	Student must have more than 9 remaining units
Success Guarantee	A notification will be sent to the student's email and the coordinator of the new course
Sub-Tasks / Scenario	
1. Student logs in to the system	
2. Student chooses to apply for an internal course transfer	
3. System will redirect the student to an online form to fill in necessary details	

4. Student fills the following details; and submits the form - student details (id, name and year) and contact details (mobile phone) - current course and proposed course to transfer to - reason for transfer
5. System will display a success message and a notification will be sent to the student's email and the coordinator of the new course
6. Student logs out of the system.
Variant
4a. Student incompletely filled in the form
4a1. Student did not fill in the form completely and submits the form
4a2. System will highlight and notify the student of the unfilled field
4b. Student enters invalid information into the form
4b1. Student enters invalid data into certain fields and submits the form
4b2. System will highlight and notify the student of the error(s) in the form
4c. Student is in final year (Less than 9 units left)
4c1. System will request confirmation from the student
4c2. Student confirms action
4c3. System generates an enrollment issue for the course coordinator and sends a notice to the student email

Actor: Student Administrator

Task 1	Set Enrolment Dates
Purpose	Changes dates for opening and closing of enrolment phases for semesters
Precondition	-
Success Guarantee	Enrolment dates are modified
Sub-Tasks / Scenario	
1. Student Administrator logs in to the system	
2. Student Administrator choose the option to set enrolment dates	
3. System displays the enrolment dates for the *Summer Term, Semester 1, Winter Term and Semester 2* semesters.	
4. Student Administrator chooses the opening and closing dates for the desired semester and confirms the selection.	
5. System will display a success message and updates the enrolment dates	
6. Student Administrator logs out of the system	
Variant	

Task 2	Manage Students
Purpose	To add/modify/remove students in the system
Precondition	-
Success Guarantee	A new student is added to the system
Sub-Tasks / Scenario	
1. Student Administrator logs in to the system	
2. Student administrator chooses to manage students	
3. System shows a list of students and with the options to add/modify/remove students	
4. Student Administrator logs out of the system	

Variant
3a. Student Administrator chooses to add students
3a1. System displays a form to fill with the option to import a list from the system
3a1a. Student Administrator chooses to add a student
3a1a1. Student Administrator enters the student details and confirms selection
3a1a2. System displays a success message and adds student to the system
3a1a1a. Student Administrator enters invalid information into the form
3a1a1a1. System displays an error message highlights the invalid entry
3a1b. Student Administrator chooses to import students
3a1b1. System will provide an empty .csv and .xls template along with appropriate instructions
3a1b2. System prompts Student Administrator for the .csv and .xls file to upload into the system
3a1b3. Student Administrator browses and selects the file from the system
3a1b4. System reads the file and shows a list of the students, highlights student id duplicates and requests for confirmation
3a1b5. Student Administrator confirms the list
3a1b6. System adds students to the system
3b. Student Administrator chooses to modify a student entry
3b1. Student Administrator searches and selects a student from the system
3b2. System displays the student's details for editing
3b3. Student Administrator makes changes and confirms changes
3b4. System displays a success message and update student's information in the system
3b4a. Student Administrator enters invalid information to the form
3b4a1. System displays an error message and highlights the invalid entry
3c. Student Administrator chooses to remove students from the system
3c1. Student Administrator searches for the student to be removed
3c2. System displays a list of students matching the search criteria

3c3. Student Administrator selects students to be removed from the list and confirms selection
3c4. System prompts Student Administrator for confirmation with a summary of the students that will be removed
3c5. Student Administrator confirms action
3c6. System moves the student into archive, logs the transaction and displays a success message.

Actor: Course Coordinator

Task 1	Manage Units
Purpose	To add/modify/remove units in the system
Precondition	-
Success Guarantee	A unit is added/modified/removed from the system

Sub-Tasks / Scenario
1. Course Coordinator logs in to the system
2. Course Coordinator chooses to manage units
3. System shows the list of units the Course Coordinator is in charge of
4. Course Coordinator logs out of the system

Variant
3a. Course Coordinator chooses to add a unit
3a1. System displays a form requesting for unit title and unit code from the Course Coordinator
3a2. Course Coordinator enters the unit title and unit code and confirms the entry
3a3. System displays a success message and adds the unit to the system
3b. Course Coordinator chooses to modify a unit
3b1. Course Coordinator chooses a unit from the list of units to modify
3b2. System displays the unit details for the Course Coordinator to modify
3b3. Course Coordinator makes changes and confirms changes
3b4. System displays a success message and updates the unit information
3c. Course Coordinator chooses to remove a unit
3c1. Course Coordinator selects a unit from the unit list to remove
3c2. System displays a message requesting confirmation of unit removal from the Course coordinator
3c3. Course Coordinator confirms action
3c4. System moves the unit into archive, logs the transaction and displays a success message.

3d. Course Coordinator chooses to import units
3d1. System will provide an empty .csv and .xls template along with appropriate instructions
3d2. System prompts Course Coordinator for the .csv and .xls file to upload into the system
3d3. Course Coordinator browses and selects the file from the system
3d4. System reads the file and shows a list of the unit details, highlights unit code duplicates and requests for confirmation
3d5. Course Coordinator confirms the list
3d6. System adds students to the system

Task 2	Manage Study Planner
Purpose	To add/remove units to the study planner of a particular semester
Precondition	-
Success Guarantee	Units are successfully added/removed from the study planner
Sub-Tasks / Scenario	
1. Course Coordinator logs in to the system	
2. Course Coordinator chooses to manage study planner	
3. System prompts Course Coordinator for the student intake	
4. Course Coordinator chooses the intake	
5. System displays the units in the selected intake	
6. Course Coordinator logs out of the system	
Variant	
3a. Course Coordinator chooses to add a unit	
3a1. System displays a list of units for the Course Coordinator to add	
3a2. Course Coordinator selects the unit(s) to add to the study planner	
3a3. System displays a success message and adds the unit to the study planner	
3b. Course Coordinator chooses to remove a unit	

3b1. Course Coordinator selects a unit from the study planner to remove
3b2. System displays a message requesting confirmation of unit removal from the Course Coordinator
3b3. Course Coordinator confirms action
3b4. System displays a success message and removes the unit from the study planner

Task 3	Manage Unit Listing
Purpose	To add/remove units to the unit listing
Precondition	-
Success Guarantee	Unit Listing is successfully modified
Sub-Tasks / Scenario	
1. Course Coordinator logs in to the system	
2. Course Coordinator chooses to manage unit listing	
3. System prompts Course Coordinator for the semester	
4. Course Coordinator chooses the semester	
5. System displays a the unit listing list	
6. Course Coordinator logs out of the system	
Variant	
5a. Course Coordinator chooses to add a unit	
5a1. System displays a list of units for the Course Coordinator to add	
5a2. Course Coordinator selects the unit(s) to add to the unit listing	
5a3. System displays a success message and adds the unit to the unit listing	
5b. Course Coordinator chooses to remove a unit	
5b1. Course Coordinator selects a unit from the unit listing to remove	
5b2. System displays a message requesting confirmation of unit removal from the Course coordinator	

5b3. Course Coordinator confirms action
5b4. System displays a success message and removes the unit from the study planner

Task 4	Resolve Student Enrolment Issues
Purpose	To allow allow Course Coordinator to approve/disapprove student enrolments
Precondition	-
Success Guarantee	The student's units are successfully approved/disapproved
Sub-Tasks / Scenario	
1. Course Coordinator logs in to the system	
2. Course Coordinator chooses to resolve student enrolment issues	
3. System displays a list of outstanding enrolment issues	
4. Course Coordinator selects an enrolment issue from the list	
5. System displays the issue details and prompts Course Coordinator for the desired action	
6. Course Coordinator give a comment and approves/disapproves the student's enrolment to the unit	
7. System displays a confirmation message on the Course Coordinator's action	
8. Course Coordinator confirms action	
9. System displays a success message and updates student's enrolment	
10. Course Coordinator logs out of the system	
Variant	
3a. The list of outstanding enrolment issues is empty	
3a1. System displays a message notifying the Course Coordinator that the list is empty	

4.0 Scenarios

Scenario 1 (Student)

For the past few months, Diabalos is a student that been confused for the past few months about the units he has enrolled in and has been told by his lecturer that he should not enrol in Software Architecture next semester rather take an elective next and do it next year as the workload next semester might be too hard for him. He tries to find the unit planner but can't find the planner in the

email anywhere so he finds it hard to enrol as there is not study planner.

Scenario 2 (Student Administrator)

Mrs Mephistopheles is student administrator who has been having a hard time trying to enrol students as when she is asked from the students to recommend which units they should enrol instead she doesn't know which ones are from the same faculty and which ones are not so basically she has no guidelines to help the students.

Scenario 3 (Coordinator)

Dr Lucifer Morningstar is a lecturer and a course coordinator at the ███████ University █ ███████████████ and has been contacted in-person by students regarding the changes in prerequisites they had to complete for the units for upcoming semester. Dr. Lucifer has to look up the units and it takes quite a lot of his time as he has lectures to teach and he often runs out of time and the students just have to wait for his email.

5.0 External Interface Requirements

5.1 User Interfaces

The user interface will apply the following design principles:
- Visibility
- Simplicity
- Responsivity
- Minimalism
- Consistency

The basic design will be made similar to the current systems in use such as Allocate+ and Blackboard for user familiarity. The interface will be responsive to all devices with different screen dimensions and resolutions.

5.2 Software Interfaces

As the Student Enrolment System is a web-based application, the system will be accessible through web browsers such as Internet Explorer, Safari, Mozilla Firefox, Google Chrome and other latest web browsers that supports webkit. In addition to that, it is accessible from any devices of different screen ratios and resolutions and also from any platforms including Windows, Android, iOS and Linux.

6.0 Non-Functional Requirements

Reliability

The systems and hardware installed should be reliable as is yielding consistent results in different experiments or statistical trials. It is also the ability of an item to perform a required function under stated conditions for a specified period of time. In the student enrolment system all the functions that are added in the system will perform with 100% accuracy. Should the system encounter any unexpected errors, it will handle them gracefully without crashing.

Availability

Availability is the ability of the system to be ready for usage. As such, the system will have an uptime of more than 95% and each day will have a downtime of no more than an hour.

Performance

The system should be fast and efficient as is the measure of how fast application application executes system tasks. The system will respond to users' requests with no unnecessary processing overhead. The system will be able to respond within a second when 1000 users are simultaneously using its service.

Usability

The application should be usable and must be user-friendly. The system is made user friendly through employing a simple and intuitive design. In addition, the system have an easily understood layout and will use soft colours to prevent eye strain. Finally, the complexity of each page will be low and have a minimalistic design.

Security

The data on the system should be secured with the implementation of SSL certificate (https), which means that all data or communication within the system will be encrypted. Different parts of the system will only be accessed to certain users. For example, a student can only view pages like viewing study planner, unit listings and managing units; and the student should not be able to view pages meant only for course coordinators or student admin like setting enrolment dates. There should be no worry attacks as the system will be part of Cloudflare community and there will be a protection against DDoS attacks.

Document Approval

This document is approved by:

_____

█████████████

(Project Client)

_____

Dr Patrick Then
(Project Supervisor)

_____

██████████████████

(Project Team Leader)