SOFTWARE REQUIREMENT SPECIFICATION (SRS)

for the

DEFERMENT OF STUDY APPLICATION SYSTEM
PYRAMID-FT27-SRS-V(1.0)

Prepared For:
Razak Faculty of Informatics and Technology, Universiti Teknologi Malaysia

Prepared by:
PYRAMID-FT27

Date created: 14/12/2023

DOCUMENT IDENTIFICATION			
SYSTEM NAME	**FORMAT**	**VERSION**	**PAGE**
DEFERMENT OF STUDY APPLICATION SYSTEM (DoSAS)	A4	1.0	OVER

DOCUMENT APPROVAL

	Name	Date
Verified by: _____ Project Leader		
Authenticated by: _____ Project Manager		
Approved by: _____ Client		

Software: Microsoft Word 2016

Archiving place: C:\ (*Where you place an archive*)

Copies available: .DOC, .PDF format and CD-ROM

REVISION HISTORY

REVISION	DESCRIPTION
A	
B	
C	
D	
E	

Ind.+ Date	A	B	C	D	E
Written by					
Verified by					
Authenticated by					
Approved by					

LIST OF EFFECTIVE PAGES

PAGES	REVISION	PAGES	REVISION

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

1 INTRODUCTION

1.1 Identification

1. System Number: DoSAS1135
2. System Name: Deferment Of Study Application System(DoSAS)

1.2 Purpose, Objectives and Scope

1.2.1 Purpose

The purpose of this document is to specify the user requirements for the Deferment of Study Application System(DoSAS) in the Advanced Informatics Department (AID), Razak Faculty of Technology and Informatics (RFTI), Universiti Teknologi Malaysia (UTM).

This document describes the required system facilities and system performance characteristics.

1.2.2 Objectives

The objectives of this document are to:

1. To define the user requirements for DoSAS.
2. To determine the scope of the system.
3. To determine system capacity and future expansion requirements.
4. To provide a basis for system development.

1.2.3 Scope

Software Interface

The function of this system will be interfacing particularly to:

1. The Student
2. The Academic Office Assistant
3. The Program Coordinator
4. The Supervisor

Software Documentation

A complete and easily understood documentation of the Deferment of Study Application System requirements will be provided to aid in the future maintenance of the system. The following deliverables will be produced:

1. Software Requirement Specification (SRS)

1.3 Definitions, Acronyms, and Abbreviations

1.3.1 Definitions of Terms

The following terms have special meanings with this document.

1. The word “shall” imply a mandatory requirement.
2. The word “should” imply a desirable requirement.
3. The word “will” imply a mandatory requirement outside the scope of this document.
4. The word “may” imply a desirable requirement outside the scope of this document.

The following terms have been defined for the context of this project:

1. Client: The individual or organization that specifies requirements for and accepts delivery of a new or modified software product and its documentation.
2. Project Deliverable: A work product to be delivered to the client. Quantities, delivery dates, and delivery locations are specified in a project agreement. Project deliverables may include the following: operational requirements, functional specifications, design documentation, source code, object code, test results, installation instructions, training aids, user’s manuals, product development tools, and maintenance documentation.
3. Software Project: The set of work activities, both technical and managerial, required to satisfy the terms and conditions of a project agreement. A software project should have specific starting and ending dates, well-defined objectives and constraints, established responsibilities, and a budget and schedule.

1.3.2 Acronyms and Abbreviations

Abbreviation	Meaning
CRUD	Create, Read, Update, and Delete
DoSAS	Deferment of Study Application System
GB	Gigabyte
IDE	Integrated Development Environment
OOD	Object Oriented Development
PHP	Hypertext Pre-processor
RAM	Random Access Memory
RFTI	Razak Faculty of Technology and Informatics
SDD	Software Design Document
SPMP	Software Project Management Plan
SQL	Structured Query Language
SRS	Software Requirement Specification
STD	Software Test Document
UML	Unified Modelling Language
UTM	Universiti Teknologi Malaysia
WBS	WorkBreakdown Structure

1.4 References

1. MSE-FT27-2024-01 User Requirement Specification for Deferment Of Study Application System
2. DoSAS Software Project Management Plan (2023), Pyramid-FT27.
3. IEEE 830- 1998 - IEEE Recommended Practice for Software Requirements Specifications.
4. Object Management Group (2010), Unified Modelling Language (OMG UML 2.3), Superstructure (10-05-05).

1.5 Overview

The purpose of the Deferment of Study Application System Requirement Specification (SRS) is to provide a detailed overview of the software product that should be developed and to guide Pyramid-FT27 team about the user requirements during the development of the

system. The overall description on the products interfaces, functions, user characteristics, rules and constraints, assumptions and dependencies will be captured in Chapter 2 of this document. The following Chapter 3 will discuss each specific requirement in detail. The last part of this document would be the appendices where all relevant sequence diagrams are illustrated.

2 OVERALL DESCRIPTION

2.1 Introduction

The DoSAS is a system developed to help students apply for study extensions. In addition, DoSAS should also have the following functions: login function, Application for extension function, Check Extension Status, Processing of applications function, View Student Details function and Approval of extensions function. The roles in DoSAS should include students, office assistants, and coordinators. DoSAS should allow students, office assistants, and coordinators to log in. Students should be able to submit a study extension application, and the office assistant and coordinator should be able to view the student's personal information and have the office assistant approve whether the student can apply, while the coordinator approves whether the student can be extended. If the office assistant or coordinator refuses the application, feedback needs to be provided to the students.

2.2 Product Perspective

DoSAS is a web-based system that uses the client-server model. It provides a simple interface for the Student to applies for deferment and view defermentand allow the academic office assistant and supervisor to view student details for those students based on the provided applies for deferment and finally allow the Program Coordinator to assign view student details for those students.

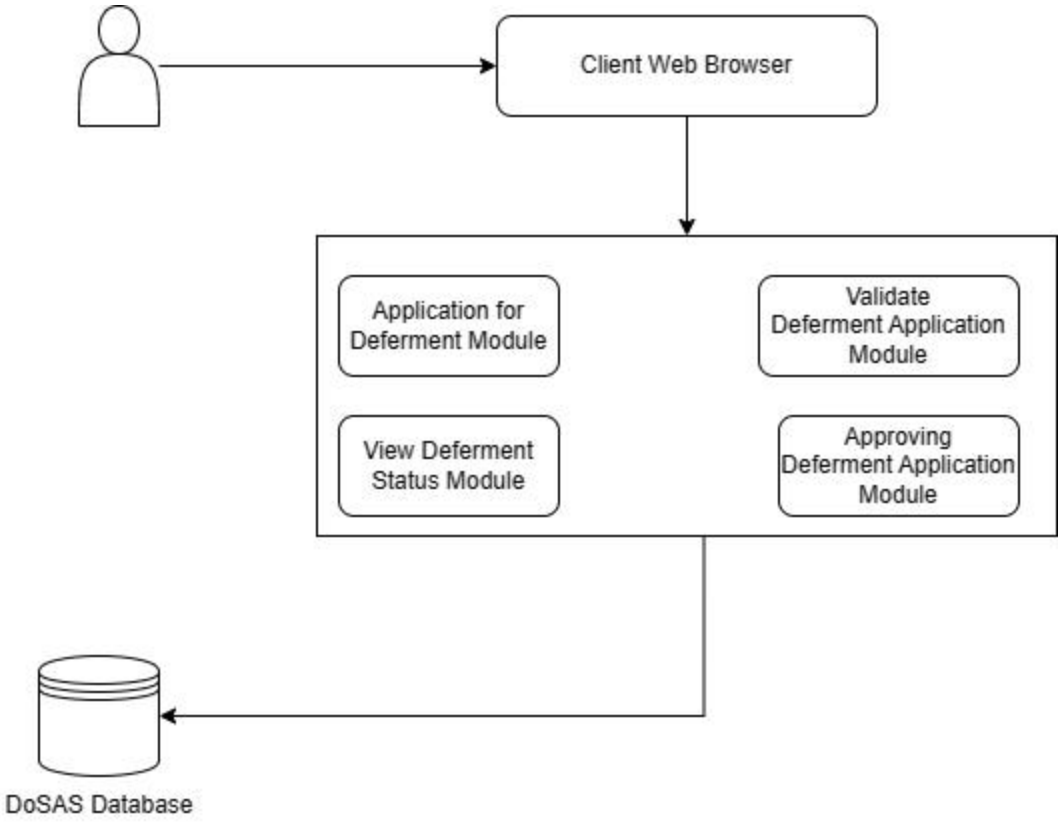

Figure 1: DoSAS Product Prospective Diagram

2.3 Product Functions

Product functions to be implemented prior to this document are:

1. Login
2. Applying for Deferment.
3. Viewing Deferment Status.
4. Approving Deferment Application.
5. Validating Deferment Application.

2.4 User Characteristics

The Users shall either be:

1. Student: Fill out the deferment application form, upload the required documents, and then submit the application.
2. Office Assistant: Review deferment requests, check completeness, and approve or reject applications.
3. Coordinator: Review deferment requests and approve or reject them.
4. Supervisor :Review deferment requests and approve or reject them.

2.5 Project constraints

The time for the development of Deferment of Study Application System (DoSAS) is 6 weeks. Therefore, the time is limited and may affect the complete fulfilments of some of the requirements.

2.6 Assumptions and Dependencies

None. No identified factors that may affect the requirements stated in the SRS.

2.7 Apportioning of Requirements

None. No requirements that may be delayed until future versions of the system.

3 SPECIFIC REQUIREMENTS

3.1 CSCI External Requirments

This section describes the functional requirements of the DoSAS in the text. **Figure 2** shows a Context Diagram which indicates the roles of the software systems:

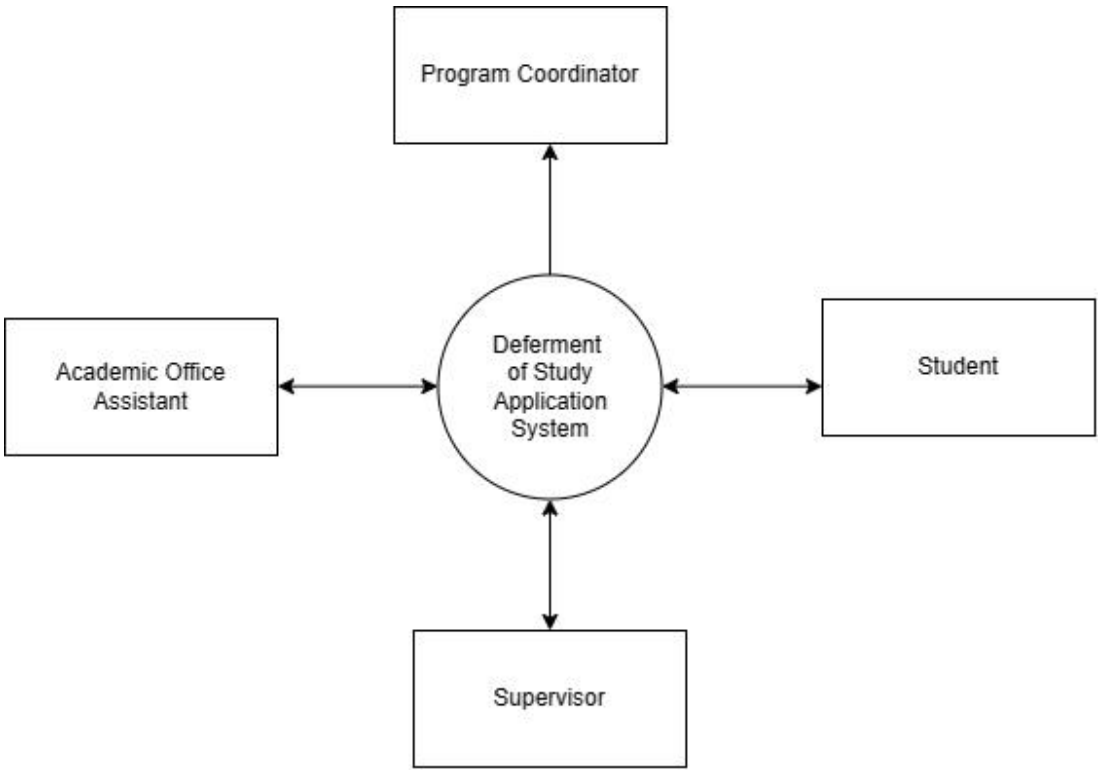

Figure 2: DoSAS Context Diagram

3.1.1 Interface:Academic Office Assistant

3.1.1.1 Identification

1. Interface identification: Academic Office Assistant
2. Interface type: Person

3.1.1.2 Description

The Academic Office Assistant is the actor who manages the extension requests.

3.1.1.3 Association

The Actor communicates with the login, View Student Details, Processing of applications,Update Application Status,Upload file and Approval file use cases

3.1.2 Interface: Supervisor

3.1.2.1 Identification

1. Interface identification: Supervisor
2. Interface type: Person

3.1.2.1 Description

The supervisor are actors who view and download information about student extensions.

3.1.2.2 Association

The actor communicates with login, Download file, View Student Details,Upload file and Approval file use cases.

3.1.3 Interface: Program Coordinator

3.1.3.1 Identification

1. Interface identification:Program Coordinator
2. Interface type: Person

3.1.3.2 Description

The program coordinator is the actor who views the student's details.

3.1.3.3 Association

The actor communicates with login, View student Details use cases.

3.1.4 Interface: Student

3.1.3.1 Identification

1. Interface identification:Student
2. Interface type: Person

3.1.3.2 Description

The student are actors who apply for deferment.

3.1.3.3 Association

The actor communicates with login , Applies for Deferment and View Deferment Status,use cases.

3.2 DoSAS Capability Requirements

The following diagram explains the whole use cases for DoSAS with three actors which are (Office assistant, Supervisor and Program coordinator).

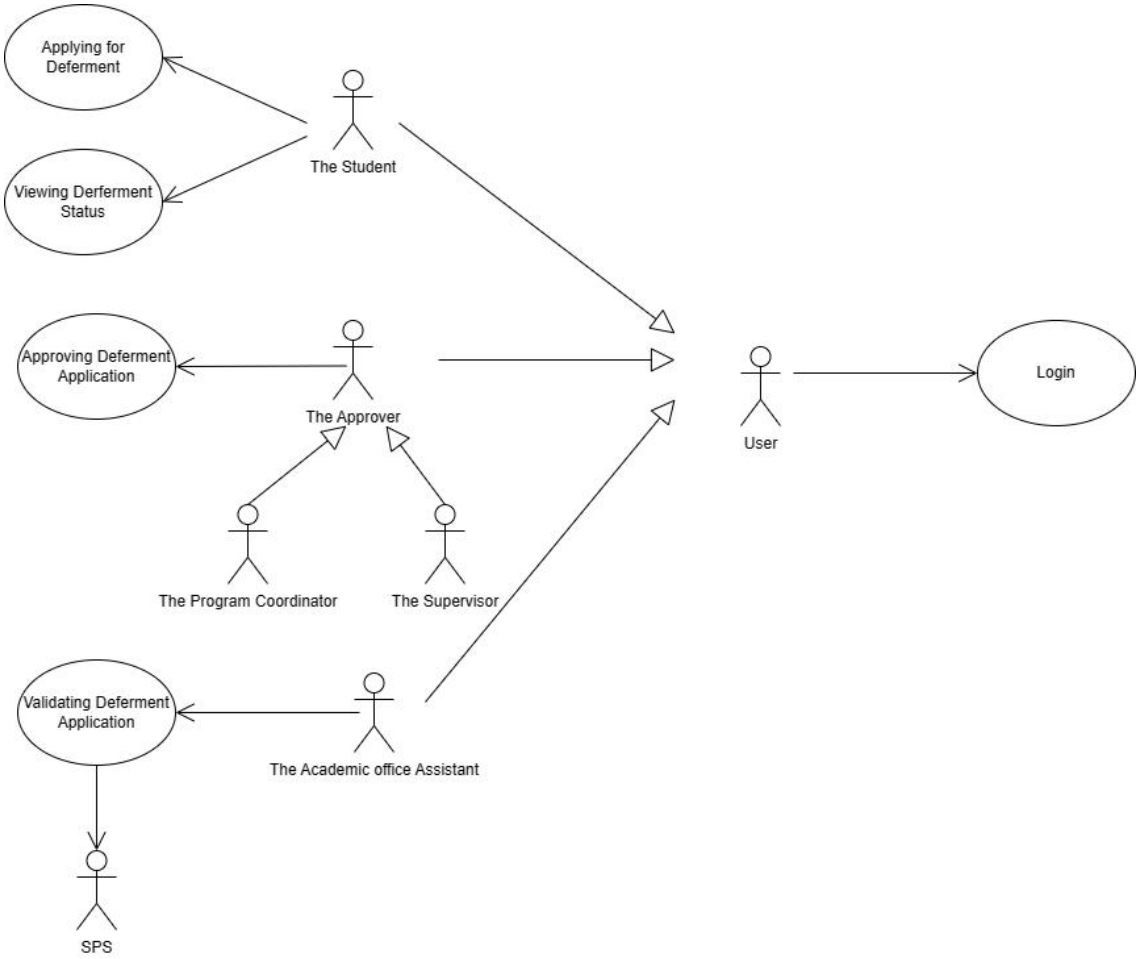

Figure 3: DoSAS Use Cases

3.2.1 Login Use Case (SRS_REQ_100)

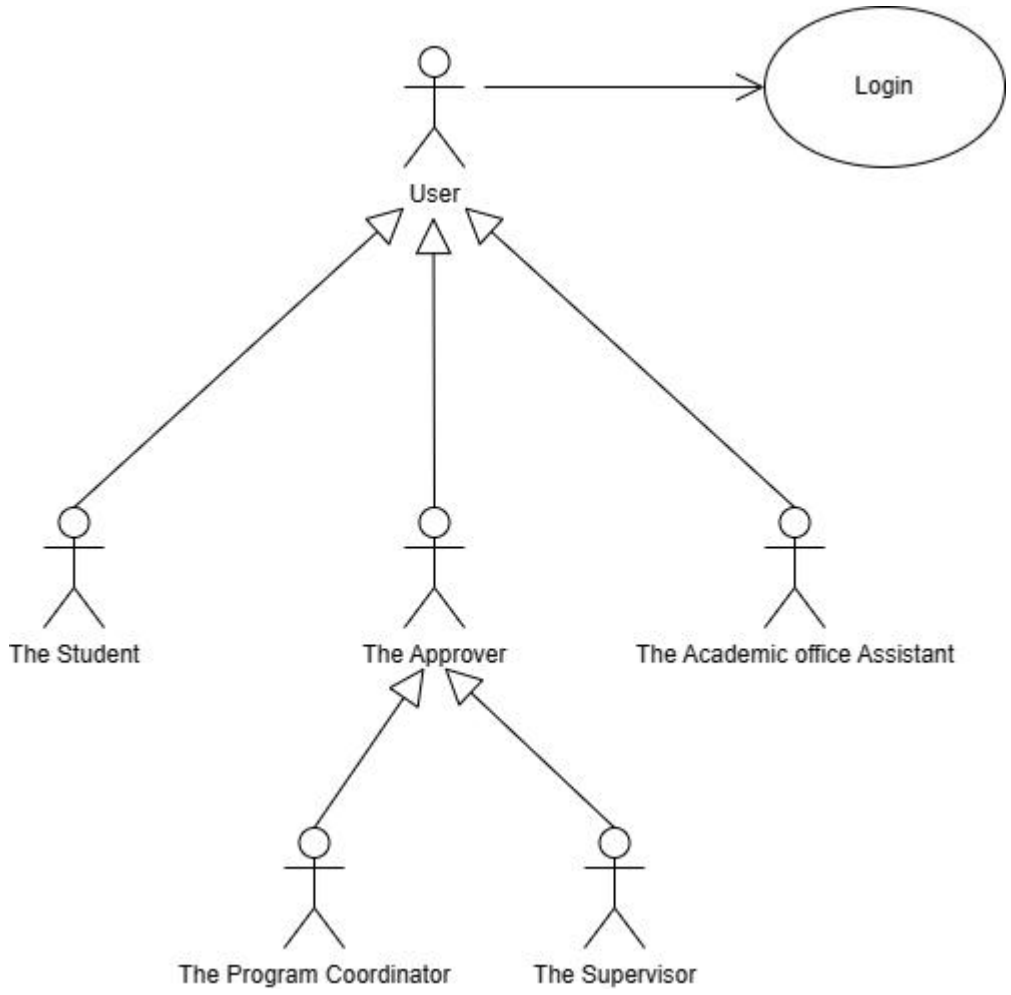

Figure 4: Login Use Case

3.2.1.1 Brief description

This use case provides capability for the user to login into the DoSAS.

3.2.1.2 Characteristic of Activation

User (Student/ Program Coordinator/ Academic Office Assistant/supervisor) initiates the use case.

3.2.1.3 Pre-condition(s)

Not Applicable

3.2.1.4 Description

3.2.1.4.1 Basic flow

1. This use case begins with the user enters the employee ID and password. **[SRS_REQ_101] [R1: Password Rules]**. If the user forgets his employee ID and password **[A1: Forgot Password]**.
2. The DoSAS validates the employee ID and password **[SRS_REQ_102]**. If the emp ID and password are incorrect **[E1: Incorrect ID and password]**.
3. The DoSAS redirects the user to the user's specific dashboard. **[SRS_REQ_103]**
4. This use case ends.

3.2.1.4.2 Alternative Flow

[A1:Forgot password]

1. The user clicks Forgot Password.**[SRS_REQ_ 104]**
2. The Reset Password form appears.**[SRS_REQ_ 105]**
3. the user enters the employee ID and submits.**[SRS_REQ_ 106]**
4. The DoSAS validates the employee ID entered.**[SRS_REQ_ 107]**
5. if the employee ID is valid, the DoSAS allows the user to reset password **[R1: Password Rules]**.**[SRS_REQ_ 108]**
6. User enters new password.**[SRS_REQ_ 109]**
7. The DoSAS validates the new password.**[SRS_REQ_ 110]**
8. If the new password is valid, the DoSAS updates the user's password.**[SRS_REQ_ 111]**
9. The DoSAS notifies the user that the password reset was successful.**[SRS_REQ_ 112]**
10. The user will return to the login page to enter the new password. **[SRS_REQ_ 113]**
11. This use case continues

[A2:First time login]

1. The DoSAS will redirect the user to the New Password page.**[SRS_REQ_ 114]**
2. The user enters the new password.**[SRS_REQ_ 115]**
3. The DoSAS validates the new password.**[SRS_REQ_ 116]**

4. If the new password is valid, the DoSAS updates the user's password.**[SRS_REQ_ 117]**
5. The DoSAS redirects the user to the Dashboard page.**[SRS_REQ_ 118]**
6. This use case ends.

3.2.1.4.3 Exception Flow

[E1: Incorrect ID and password]

1. The DoSAS notifies the user that the employee ID or password is incorrect.**[SRS_REQ_ 119]**
2. The user has two more attempts. The DoSAS notifies the user of the failure and prompts the user to enter the user name or password again.**[SRS_REQ_ 120]**
3. If the user fails in three attempts, continue with "Forgot Password".**[SRS_REQ_ 121]**
4. The use case continues.

3.2.1.5 Post conditions

The user login was successful.

3.2.1.6 Rule(s) and Constraint(s)

[R1: Password Rules]

1. The password consist of 8-16 alphanumeric characters.**[SRS_REQ_122]**
2. Incorrect password message "INCORRECT COMBINATION" **[SRS_REQ_123]**

3.2.1.7 GUI

Not applicable

3.2.2 Applying for Deferment Use Case (SRS_REQ_200)

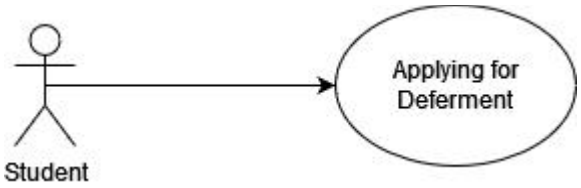

Figure 5: Applying for Deferment Use Case

3.2.2.1 Brief Description

This use case provides capabilities for the Student to apply for study deferment.

3.2.2.2 Characteristics of Activation

Only students can activate this use case

3.2.2.3 Pre-condition(s)

The student is logged in to the DoSAS.

3.2.2.4 Description

3.2.2.4.1 Basic flow

1. This use case begins when the Student selects deferment either for local or international **[SRS_REQ_201]**.
2. The DoSAS displays a form specific to the deferment type **[SRS_REQ_202]**.
3. The Student fills up these following details **[SRS_REQ_203]**:

a.Student Name

b.Student IC/ISID

c.Student Matric ID

d.Program Code (PhD, MPhil or DSE)

e.Faculty

f.Program Name

g.Student's Current Semester

h.Main Supervisor

i.Co-Supervisor

j.Nationality

k.Proposal Defense (PD) Completion Status

l.Notis Hantar Tesis (NHT) Completion Status

m.Student Deferment History

4. The Student submits the form **[SRS_REQ_204]**.
5. The DoSAS validates and stores the form **[SRS_REQ_205]**.
6. The Students download the submitted form **[SRS_REQ_206]**.

7. The Students upload the signed form **[SRS_REQ_207]**.
8. This use case ends.

3.2.1.4.2 Alternative Flow

Not applicable.

3.2.1.4.3 Exception Flow

Not applicable.

3.2.2.5 Postconditions

The Student successfully applied the deferment application.

3.2.2.6 Rule(s) and Constraint(s)

Not applicable.

3.2.2.7 GUI

Not applicable.

3.2.3 Viewing Deferment Status Use Case (SRS_REQ_300)

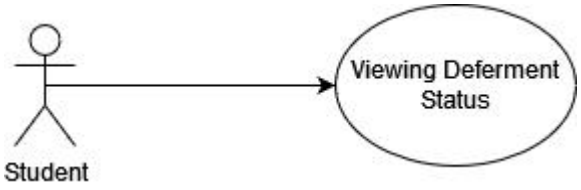

Figure 6: Viewing Deferment Status Use Case

3.2.3.1 Brief Description

This use case provides capabilities for the Student to view the deferment application.

3.2.3.2 Characteristics of Activation

The Student initiates the use case.

3.2.3.3 Pre-condition(s)

1. The Student has logged in.
2. The Student dashboard is active.

3.2.3.4 Description

3.2.3.4.1 Basic flow

1. This use case begins when the Student selects view deferment status **[SRS_REQ_301]**.
2. The DoSAS retrieves all the Student deferment records up till the most recent application **[SRS_REQ_302]**.
3. The Student shall view the application status either accepted or rejected **[SRS_REQ_303]**. If the application is rejected **[A1: Resubmitting the Application]**.
4. This use case ends.

3.2.1.4.2 Alternative Flow

[A1: Resubmitting the Application]

1. The Student provides the missing information or corrects the incorrect information respectively **[SRS_REQ_304]**.
2. The Student submits the edited application for an approval **[SRS_REQ_305]**.
3. The DoSAS stores the application and notifies the Student about the duration of approval **[SRS_REQ_306]**.
4. This use case continues.

3.2.1.4.3 Exception Flow

Not applicable.

3.2.3.5 Postconditions

The student able to update his application when necessary.

3.2.3.6 Rule(s) and Constraint(s)

Not applicable.

3.2.2.7 GUI

Not applicable.

3.2.4 Approving Deferment Application Use Case (SRS_REQ_400）

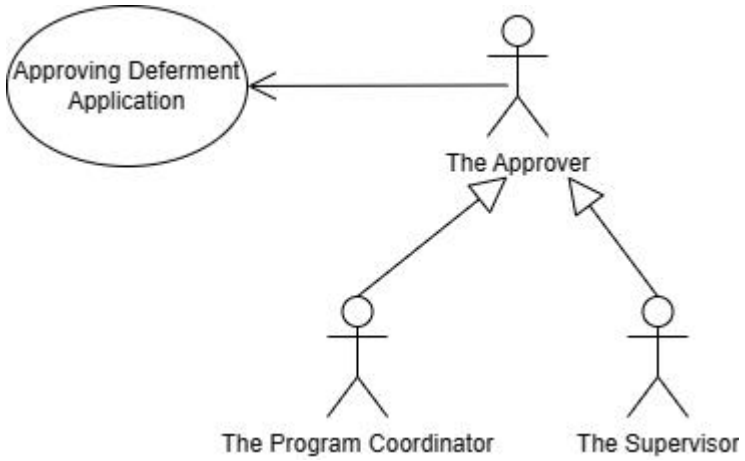

Figure 7: Approving Deferment Application Use Case

3.2.4.1 Brief description

This use case provides capabilities for the approver to approve the deferment application.

3.2.4.2 Characteristic of Activation

The approver(Program Coordinator/Supervisor) initiates the use case.

3.2.4.3 Pre-condition(s)

1. The approver has logged in.
2. The approver dashboard is active.

3.2.4.4 Description

3.2.4.4.1 Basic flow

1. This use case starts with the approver viewing an deferment application **[SRS_REQ_401]**.
2. The DoSAS lists the deferment application that have been approved by the approver and the deferment application that need to be approved **[SRS_REQ_402]**.
3. The approver selects approved or unapproved extension requests and DoSAS displays the extension request history for the selected students **[SRS_REQ_403].**
4. The approver checks the information and signature of the deferment application for accuracy and correctly clicks the Agree button **[SRS_REQ_404]**. If incorrect, click the **[E1:Reject]** button.
5. The DoSAS stores the form **[SRS_REQ_405]**.
6. The DoSAS supports multiple approvals per session **[SRS_REQ_406]**.
7. This use case ends.

3.2.4.4.2 Alternative Flow

Not applicable

3.2.4.4.3 Exception Flow

[E1: Reject]

1. The approver will leave some remarks on the form of why the application is left without an approval **[SRS_REQ_407]**.
2. The approver submits the form **[SRS_REQ_408]**.
3. The DoSAS saves the form **[SRS_REQ_409]**.
4. This use case continues.

3.2.4.5 Post conditions

The Program Coordinator approves some deferment application.

3.2.4.6 Rule(s) and Constraint(s)

Not applicable

3.2.4.7 GUI

Not applicable

3.2.5 Validating Deferment Application Use Case (SRS_REQ_500）

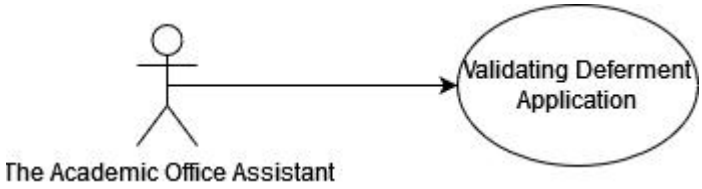

Figure 8: Validating Deferment Application Use Case

3.2.5.1 Brief description

This use case provides capabilities for the Academic Office Assistant to validate the deferment application.

3.2.5.2 Characteristic of Activation

The Academic Office Assistant initiates the use case.

3.2.5.3 Pre-condition(s)

1. The Academic Office Assistant has logged in.
2. The Academic Office Assistant dashboard is active.

3.2.5.4 Description

3.2.5.4.1 Basic flow

1. This use case begins when the Academic Office Assistant views the new deferment application **[SRS_REQ_501]**.
2. The DoSAS displays all the application information **[SRS_REQ_502]**.
3. The Academic Office Assistant validates all the necessary information is correct **[SRS_REQ_503]**. If the Academic Office Assistant finds that the necessary information is missing **[E1: Missing Information]**.

4. The Academic Office Assistant submits the application to the DoSAS for an approval **[SRS_REQ_504]**.
5. The DoSAS emails the application to the SPS for the approval **[SRS_REQ_505].**
6. This use case ends.

3.2.4.5.2 Alternative Flow

Not applicable

3.2.5.4.3 Exception Flow

[E1: Missing Information]

1. The Academic Office Assistant will put some remarks about the missing piece of information **[SRS_REQ_506]**.
2. The Academic Office Assistant labels the application status with rejected remark **[SRS_REQ_507]**.
3. The DoSAS updates the information **[SRS_REQ_508]**.
4. This use case ends.

3.2.4.5 Post conditions

The Academic Office Assistant validates the deferment application.

3.2.4.6 Rule(s) and Constraint(s)

Not applicable

3.2.4.7 GUI

Not applicable

3.3 System Internal Interface

3.3.1 CSCI Data Element Requirements

3.3.1.1 Boundary Class

3.3.1.1.1 LoginPage

Responsibility : Allows users to log in and have access to the DoSAS

Received Events : Received users login username and password.

Sent Events : Send the username and password for authentication.

3.3.1.1.2 Dashboard

Responsibility : Different users can navigate to other pages

Received Events : Users click on a function.

Sent Events : Transfer users to other pages.

3.3.1.1.3 ForgotPasswordPage

Responsibility : Responsible for allowing users to reset their password.

Received Events : Receive the new password from the user.

Sent Events : Send the new password to be updated.

3.3.1.1.4 NewPasswordPage

Responsibility : Responsible for allowing users to reset their passwords the first time they log in.

Received Events : Receive the new password from the user.

Sent Events : Send the new password to be updated.

3.3.1.1.5 DefermentSelectionPage

Responsibility : Responsible for displaying the relevant options for selecting or specifying deferment details.

Received Events : Receive students to choose the deferment option.

Sent Events : Send student submission deferment form.

3.3.1.1.6 DefermentFormPage

Responsibility : Responsible for completing deferment application information, and downloading and uploading deferment applications.

Received Events : Receive information on student deferments..

Sent Events : Sending Student Submitted Deferment Forms and Uploads Signed Forms.

3.3.1.1.7 ViewDefermentStatusPage

Responsibility : Responsible for displaying and updating the student deferment request details and the current status to the user.

Received Events : Receive user login and access to "ViewDefermentStatusPage".

Sent Events : Sends the user to view and update the deferment request form and retrieve the current status of the student's deferment request.

3.3.1.1.8 StudentDefermentApplicationListPage

Responsibility : Responsible for displaying a list and detailed information and history of student extension requests.

Received Events : Receive list and detail information and history of student deferment requests.

Sent Events : Send approved deferment application form

3.3.1.2 Control Class

3.3.1.2.1 UserController

Responsibility : Responsible for validating user information.

Received Events : Receiving user information.

Sent Events :Verify the User information from the database.

3.3.1.2.2 DefermentControl

Responsibility : Responsible for receiving deferment requests, approvals, and denials for processing.

Received Events : Receive new deferment of Study requests from users, process deferment requests, review and approve/deny deferment requests.

Sent Events :Send information on extensions saved and updated in the database, and notifies the results of deferment requests.

3.3.1.3 Entity Class

3.3.1.3.1 Database

Responsibility :Manage table in the database

Received Events :Receive validated Data for verification,update or insertion

Sent Events :Send validation, insertion, update results.

3.4 Other Requirements

3.4.1 Reliability

The DoSAS downtime shall not exceed 6 hours (optional).

3.4.2 Maintainability

The DoSAS shall be designed and developed using a framework that allows easy maintainability.

3.4.3 Performance

The DoSAS system shall be able to access anywhere and anytime.

3.4.4 Security

Access to operate the functions available to the actors will be restricted by using the correct password.

3.4.5 Expandability

The DoSAS shall be designed for further module integration and enhancement.

3.4.6 Traceability Requirements

SN	SOURCE	REQUIREMENT	ID DESCRIPTION	SECTION
Login Use Case (SRS_REQ_100)				
1	URS	SRS_REQ_101	This use case begins with the user enters the emp ID and password.	3.2.1.4
2	URS	SRS_REQ_102	The DoSAS validates the emp ID and password.	3.2.1.4
3	URS	SRS_REQ_103	The DoSAS redirects the user to the user's specific dashboard.	3.2.1.4
4	URS	SRS_REQ_104	The user clicks Forgot Password.	3.2.1.4
5	URS	SRS_REQ_105	The Reset Password form appears.	3.2.1.4
6	URS	SRS_REQ_106	The user enters the employee ID and submits.	3.2.1.4
7	URS	SRS_REQ_107	The DoSAS validates the employee ID entered.	3.2.1.4
8	URS	SRS_REQ_108	if the employee ID is valid, the DoSAS allows the user to reset password .	3.2.1.4

9	URS	SRS_REQ_109	User enters new password.	3.2.1.4
10	URS	SRS_REQ_110	The DoSAS validates the new password.	3.2.1.4
11	URS	SRS_REQ_111	If the new password is valid, the DoSAS updates the user's password.	3.2.1.4
12	URS	SRS_REQ_112	The DoSAS notifies the user that the password reset was successful.	3.2.1.4
13	URS	SRS_REQ_113	The user will return to the login page to enter the new password.	3.2.1.4
14	URS	SRS_REQ_114	The DoSASwill redirect the user to the New Password page.	3.2.1.4
15	URS	SRS_REQ_115	The user enters the new password.	3.2.1.4
16	URS	SRS_REQ_116	The DoSAS validates the new password.	3.2.1.4
17	URS	SRS_REQ_117	If the new password is valid, the DoSAS updates the user's password.	3.2.1.4
18	URS	SRS_REQ_118	The DoSAS redirects the user to the Dashboard page.	3.2.1.4
19	URS	SRS_REQ_119	The DoSAS notifies the user that the employee ID or password is incorrect.	3.2.1.4
20	URS	SRS_REQ_120	The user has two more attempts. The DoSAS notifies the user of the failure and prompts the user to enter the user name or password again.	3.2.1.4
21	URS	SRS_REQ_121	If the user fails in three attempts, continue with "Forgot Password".	3.2.1.4
Appling for Deferment Use Case (SRS_REQ_200)				
22	URS	SRS_REQ_201	This use case begins when the Student selects deferment either for local or international	3.2.1.4
23	URS	SRS_REQ_202	The DoSAS displays a form specific to the deferment type	3.2.1.4
24	URS	SRS_REQ_203	The Student fills up these following details	3.2.1.4
25	URS	SRS_REQ_204	The Student submits the form	3.2.1.4
26	URS	SRS_REQ_205	The DoSAS validates and stores the form	3.2.1.4
27	URS	SRS_REQ_206	The Students download the submitted form	3.2.1.4
	URS	SRS_REQ_207	The Students upload the signed form	3.2.1.4
Viewing Deferment Status Use Case (SRS_REQ_300)				
28	URS	SRS_REQ_301	This use case begins when the Student selects view deferment	

			status	3.2.1.4
29	URS	SRS_REQ_302	The DoSAS retrieves all the Student deferment records up till the most recent application	3.2.1.4
30	URS	SRS_REQ_303	The Student shall view the application status either accepted or rejected	3.2.1.4
32	URS	SRS_REQ_304	The Student provides the missing information or corrects the incorrect information respectively	3.2.1.4
33	URS	SRS_REQ_305	The Student submits the edited application for an approval	3.2.1.4
34	URS	SRS_REQ_306	The DoSAS stores the application and notifies the Student about the duration of approval	3.2.1.4
Approving Derferment Application Use Case (SRS_REQ_400）				
35	URS	SRS_REQ_401	This use case starts with the approver viewing an deferment application	3.2.1.4
36	URS	SRS_REQ_402	The DoSAS lists the deferment application that have been approved by the approver and the deferment application that need to be approved	3.2.1.4
37	URS	SRS_REQ_403	If the approver selects Approved deferment application, DoSAS displays the deferment application history for the selected student. If the approver selects an unapproved deferment application, DoSAS displays the selected student's deferment application history and detail information for the current application	3.2.1.4
38	URS	SRS_REQ_404	The approver checks the information and signature of the deferment application for accuracy and correctly clicks the Agree button4.. If incorrect, click the [E1:Reject] button.	3.2.1.4
39		SRS_REQ_405	The DoSAS stores the form.	
40	URS	SRS_REQ_406	The DoSAS supports multiple approvals per session	3.2.1.4
41	URS	SRS_REQ_407	The approver will leave some remarks on the form of why the application is left without an approval	3.2.1.4
42	URS	SRS_REQ_408	The approver submits the form	3.2.1.4
43	URS	SRS_REQ_409	The DoSAS saves the form	3.2.1.4
Validating Deferment Application Use Case(SRS_REQ_500）				
44	URS	SRS_REQ_501	This use case begins when the Academic Office Assistant	

			views the new deferment application	3.2.1.4
45	URS	SRS_REQ_502	The DoSAS displays all the application information	3.2.1.4
46	URS	SRS_REQ_503	The Academic Office Assistant validates all the necessary information is correct	3.2.1.4
47	URS	SRS_REQ_504	The Academic Office Assistant submits the application to the DoSAS for an approval	3.2.1.4
48	URS	SRS_REQ_505	The DoSAS emails the application to the SPS for the approval	3.2.1.4
49	URS	SRS_REQ_506	The Academic Office Assistant will put some remarks about the missing piece of information	3.2.1.4
50	URS	SRS_REQ_507	The Academic Office Assistant labels the application status with rejected remark	3.2.1.4
51	URS	SRS_REQ_508	The DoSAS updates the information	3.2.1.4

Table 1:Traceability Matrix

3.4.7 Design Constraints

Object-oriented development (OOD) techniques with UML notation shall be applied to the project.

APPENDICES

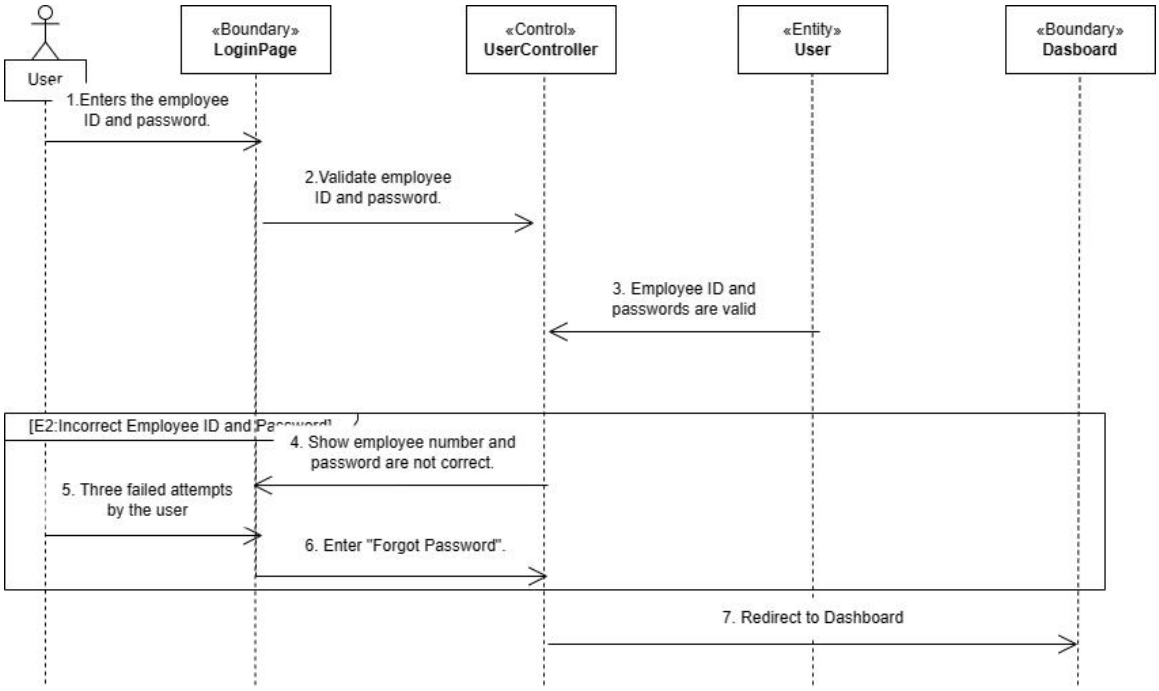

Figure 9: Login basic flow Sequence Diagram

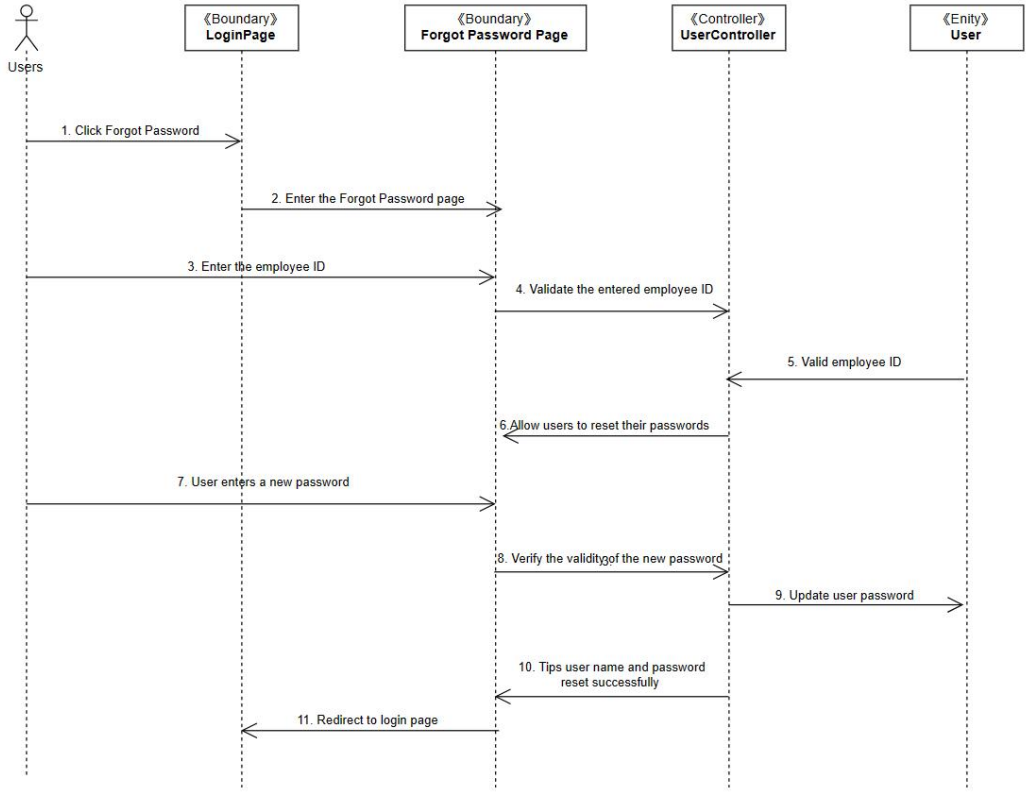

Figure 10: Login A1 Flow Sequence Diagram

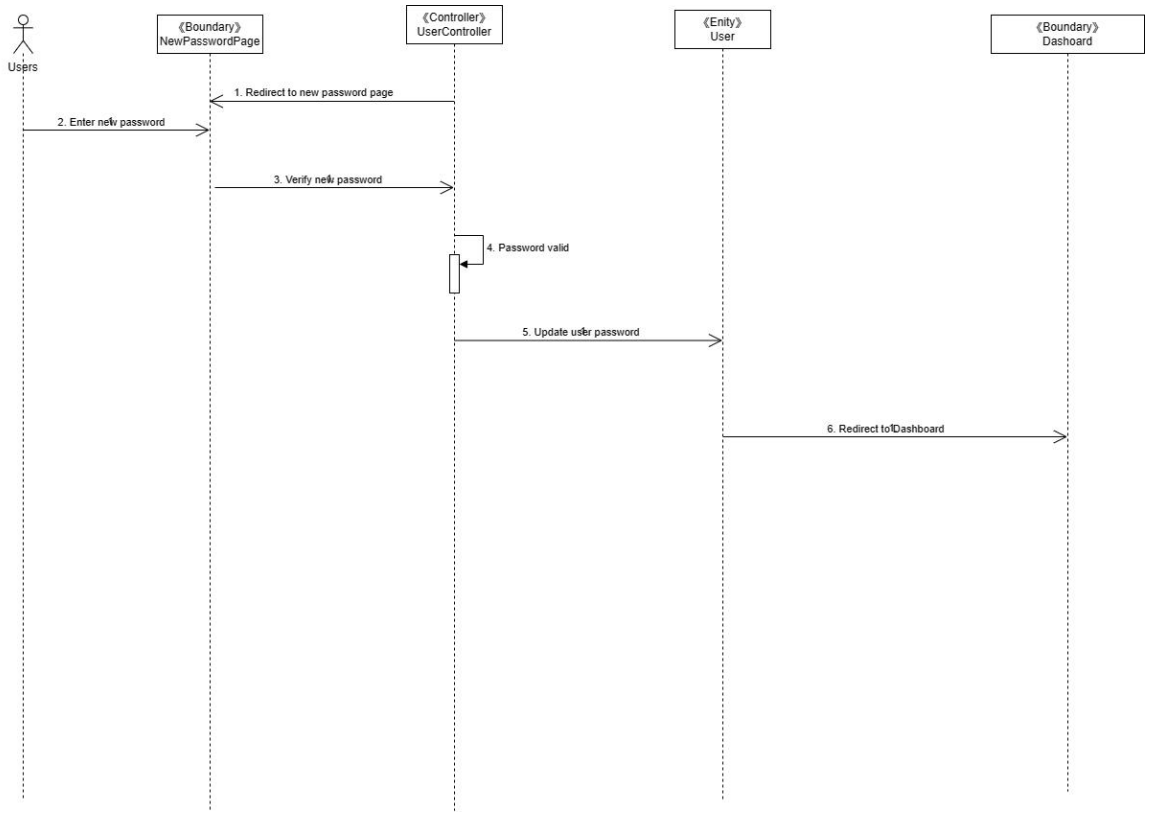

Figure 11: Login A2 Flow Sequence Diagram

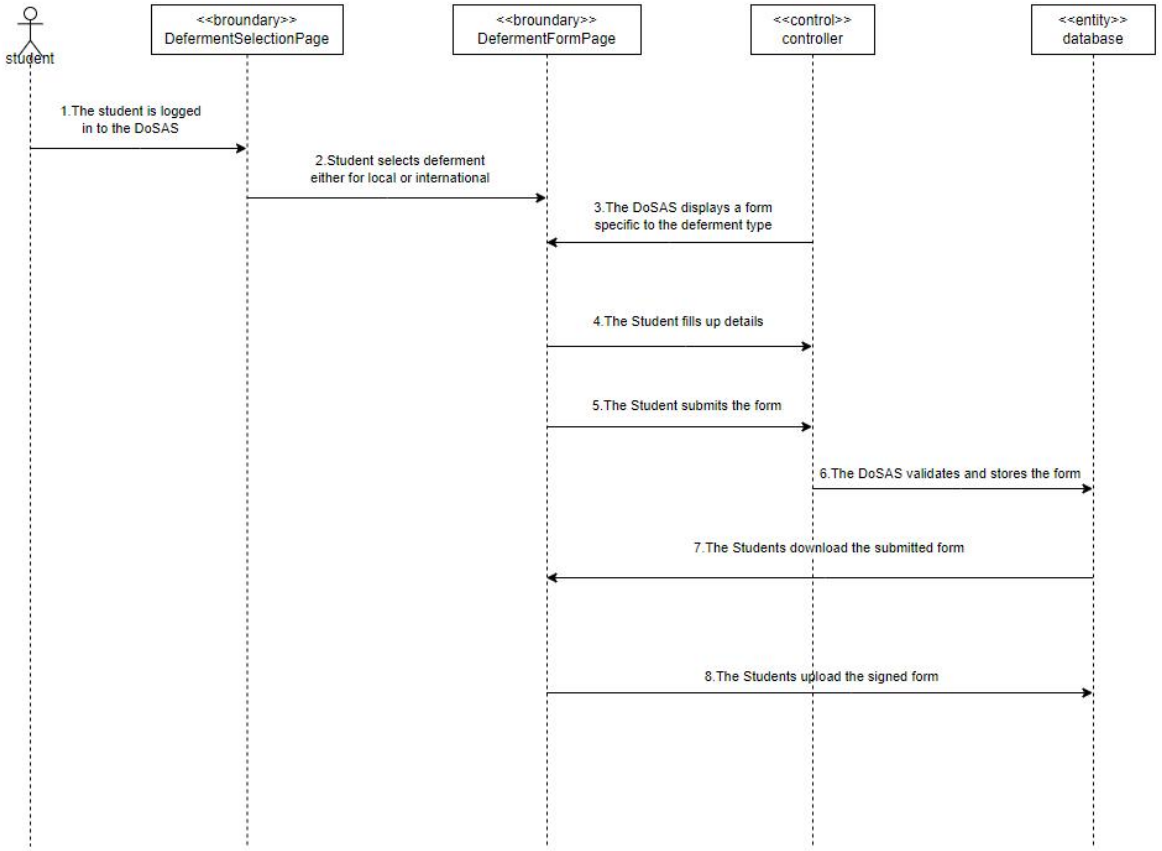

Figure 12: Applying for Deferment Basic Flow Sequence Diagram

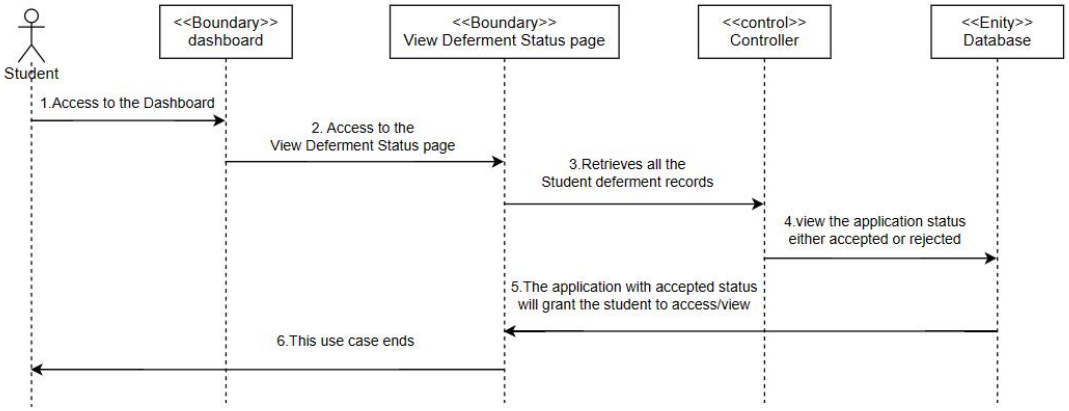

Figure 13: Viewing Deferment Status Basic Flow Sequence Diagram

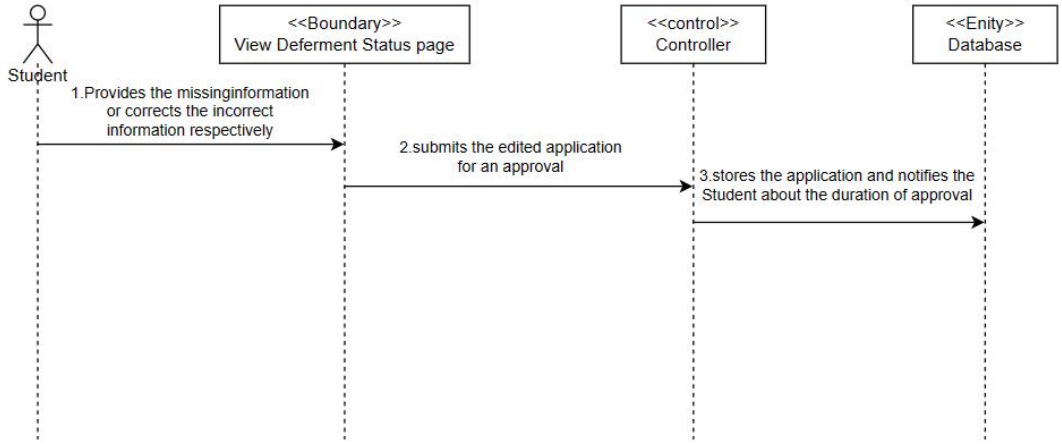

Figure 14: Viewing Deferment Status A1 Flow Sequence Diagram

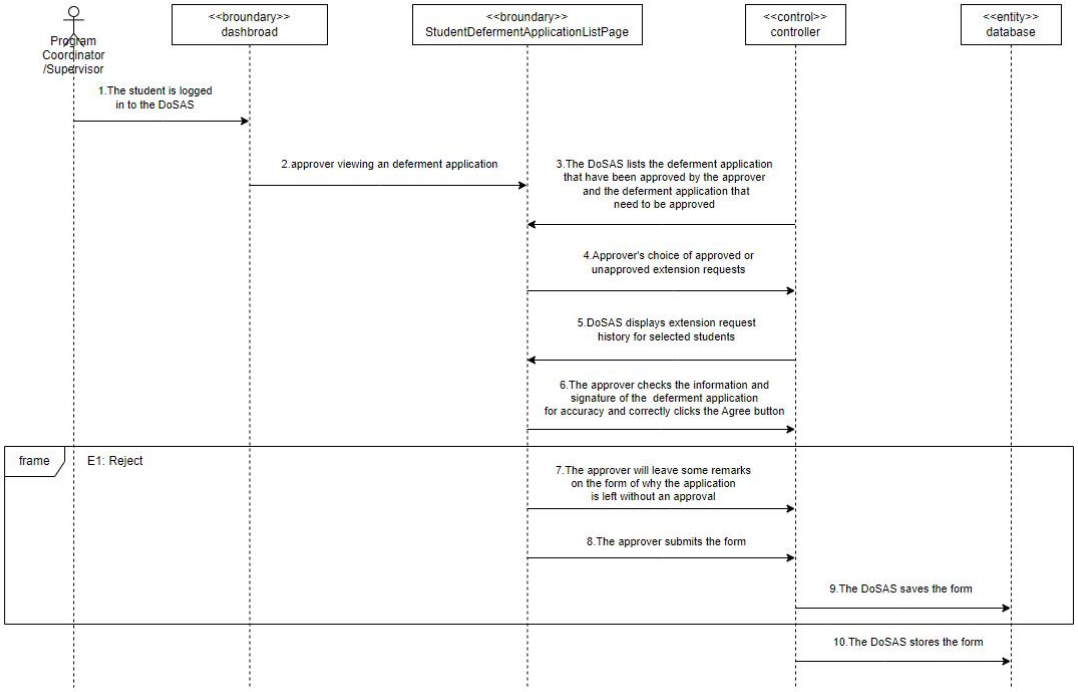

Figure 15: Approving Deferment Application Basic Flow Sequence Diagram

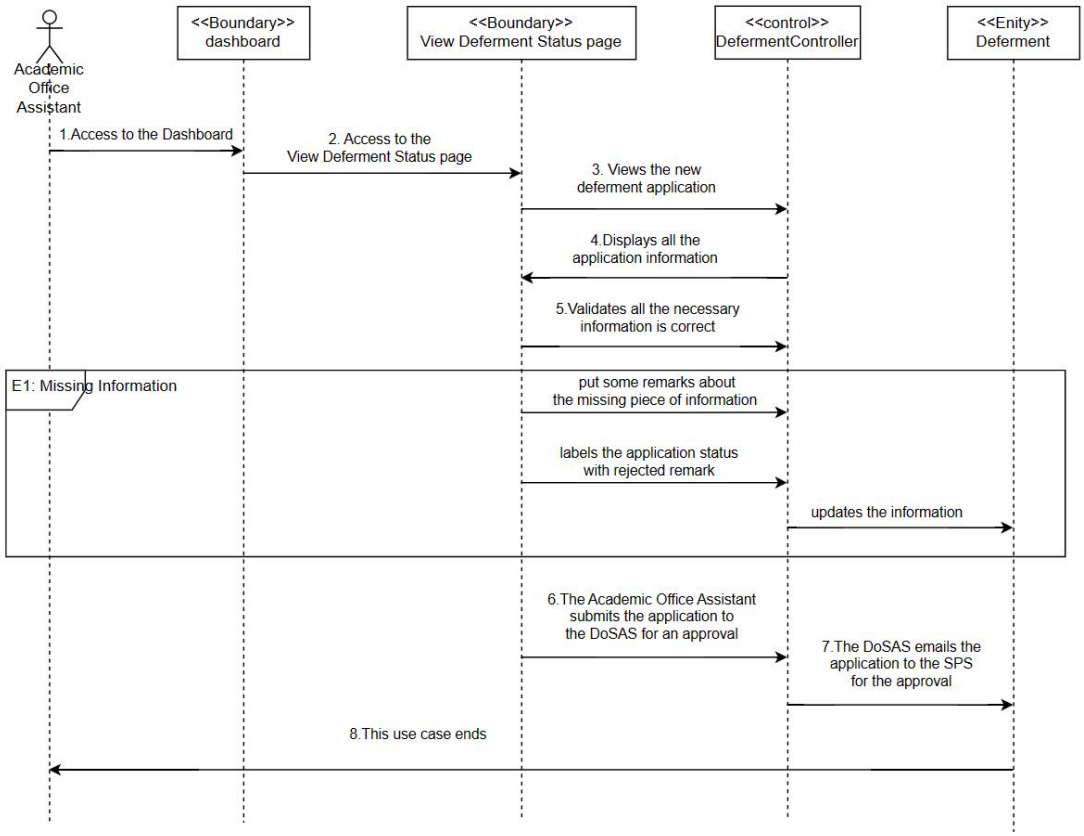

Figure 16: Validating Deferment Application Basic Flow Sequence Diagram

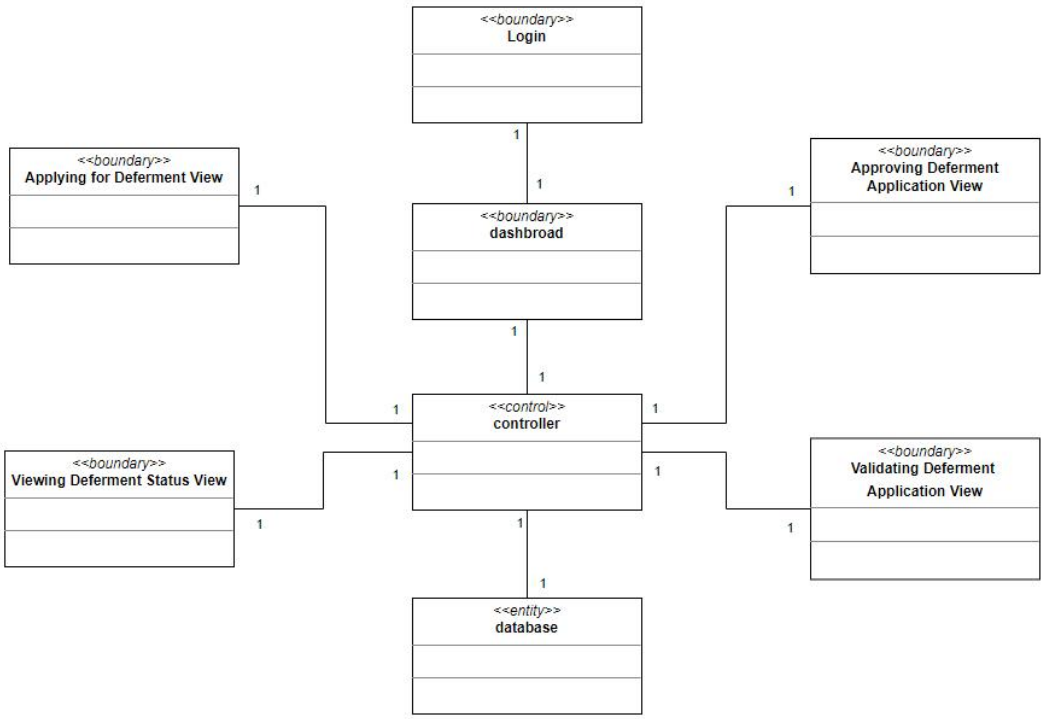

Figure 17:Class Diagram