

---

# **Software Requirements and Design Specification**

for

## **OneStop System Degree Issuance Module**

**Version 1.0 approved**

**Prepared by Ahmad Farhan**

**Software Engineering Deliverable 3**

**Group 13 Section A**

**1st May 2024**

**Group Members:**

Ahmad Farhan – i211366

Hasan Kamal – i210694

Fatima Mazhar – i210504

**Submitted to:**

Javaria Imtiaz



# Table of Contents

<b>Table of Contents .....</b>	<b>i</b>
<b>Revision History .....</b>	<b>iv</b>
<b>1. Introduction .....</b>	<b>1</b>
1.1 Purpose .....	1
1.2 Document Conventions .....	1
1.3 Intended Audience and Reading Suggestions .....	1
1.4 Problem Statement .....	2
1.5 Product Scope .....	2
1.6 References .....	2
<b>2. Overall Description .....</b>	<b>3</b>
2.1 Product Perspective .....	3
2.2 Product Functions .....	3
2.3 User Classes and Characteristics .....	4
2.4 Operating Environment .....	5
2.5 Design and Implementation Constraints .....	5
2.6 User Documentation .....	6
2.7 Assumptions and Dependencies .....	6
<b>3. External Interface Requirements .....</b>	<b>7</b>
3.1 User Interfaces .....	7
3.2 Hardware Interfaces .....	7
3.3 Software Interfaces .....	8
3.4 Communications Interfaces .....	8
<b>4. System Features .....</b>	<b>9</b>
4.1 Degree Issuance Request Submission .....	9
4.2 Complaint Form Submission .....	10
4.3 Query Progress Tracking .....	11
4.4 Receipt of Digital Documents .....	11
4.5 Notifications for Degree Collection .....	12
4.6 Student Account Registration .....	12
4.7 Payment of Degree Issuance Request Fees .....	13

4.8	Status Updates Notification .....	13
4.9	Degree Issuance Request Notification - Admin .....	14
4.10	Token Generation with Time Estimates.....	15
4.11	Degree Issuance Upon Approval.....	15
4.12	Request Management .....	16
4.13	Transcript Generation .....	17
4.14	Handling Request Objections.....	17
4.15	Automatic List Updates .....	18
4.16	Filter Requests by Day .....	19
4.17	View Pending Requests.....	19
4.18	View Processed Requests .....	20
4.19	View Processing Metrics .....	20
4.20	Active Progress Tracking - Director .....	21
4.21	Request Processing Timestamps.....	21
4.22	Degree Issuance Request Notification - FYP .....	22
4.23	Request Evaluation and Decision – FYP\ .....	22
4.24	Verification of Outstanding Fees\.....	23
4.25	Requests Evaluation and Decision - Finance .....	24
4.26	Degree Issuance Request Notification – Finance .....	24
<b>5.</b>	<b>Other Nonfunctional Requirements .....</b>	<b>25</b>
5.1	Performance Requirements .....	25
5.2	Safety Requirements .....	25
5.3	Security Requirements .....	25
5.4	Software Quality Requirements.....	25
5.5	Business Rules .....	25
<b>6.</b>	<b>Design .....</b>	<b>27</b>
6.1	Architecture.....	27
6.2	Use Case Diagram.....	29
6.3	Activity Diagram .....	30
6.4	Sequence Diagram.....	33
6.5	Class Diagram .....	35
<b>7.</b>	<b>Product Backlog.....</b>	<b>36</b>
7.1	User Story S-01 .....	36

7.2	User Story S-02 .....	36
7.3	User Story S-03 .....	37
7.4	User Story S-04 .....	37
7.5	User Story: S-05 .....	38
7.6	User Story: S-06 .....	38
7.7	User Story: S-07 .....	39
7.8	User Story: S-08 .....	39
7.9	User Story: A-01 .....	40
7.10	User Story: A-02 .....	40
7.11	User Story: A-03 .....	41
7.12	User Story: A-04 .....	41
7.13	User Story: A-05 .....	42
7.14	User Story: A-06 .....	42
7.15	User Story: A-07 .....	43
7.16	User Story: D-01 .....	43
7.17	User Story: D-02 .....	44
7.18	User Story: D-03 .....	44
7.19	User Story: D-04 .....	45
7.20	User Story: D-05 .....	45
7.21	User Story: Y-01 .....	46
7.22	User Story: Y-02: .....	46
7.23	User Story: Y-03 .....	47
7.24	User Story: F-01 .....	47
7.25	User Story: F-02 .....	48
7.26	User Story: F-03 .....	48
<b>8.</b>	<b>Software Testing.....</b>	<b>49</b>
8.1	Login Page .....	49
8.2	Issuance Form .....	50
8.3	Complaint Form .....	52
8.4	Payments .....	53
8.5	Objections to Requests .....	54

<b>Appendix A: Glossary .....</b>	<b>55</b>
<b>Appendix B: Analysis Models .....</b>	<b>56</b>
1. Github.....	56
2. Trello.....	56
3. Sprint Burndown Chart.....	59
4. Work Breakdown Structure.....	61
5. Gantt Chart.....	62
<b>Appendix C: To Be Determined List.....</b>	<b>62</b>

## Revision History

Name	Date	Reason For Changes	Version
		=	

# **1. Introduction**

## **1.1 Purpose**

This Software Requirements Specification (SRS) document outlines the requirements for the development of Degree Issuance Module within the Onestop Student Service Center System (version 1.0) at FAST Islamabad Campus. The main objective of the Onestop Services Centre is to provide assistance to students at a single doorstep. The Degree Issuance Module aims to streamline and automate the degree issuance process, which is currently manual, and paper based. By digitizing this process, the system will facilitate timely graduation, ensure accuracy in academic records, and improve operational efficiency for both students and administration.

This document specifies the functional and non-functional requirements, intended to guide the development team in building the software according to the stakeholders' needs and expectations. Stakeholders, developers, and testers can refer to this document as a reference throughout the project lifecycle.

## **1.2 Document Conventions**

This SRS document adheres to the following formatting conventions:

- SRS is documented according to IEEE 830 standard format.
- Priority levels are indicated using the prioritization scales technique (numbers 1-5).
- Requirements traceability is maintained through unique identifiers assigned to each requirement.

## **1.3 Intended Audience and Reading Suggestions**

This document is intended for various stakeholders involved in the development and management of the software, including:

- System Engineers
- Project Managers
- System Customers
- Quality Assurance/Testers
- Business Analysts

## 1.4 Problem Statement

The Problem of manual and paper-based degree issuance effects the students and administration of FAST Islamabad. The impact of which is:

- Hindrances in timely graduation of students
- Inaccuracies in academic records
- Operational inefficiencies

A successful solution would be a comprehensive system that facilitates the processing and issuance of degrees for undergraduate and graduate programs. The students will be able to submit degree issuance requests and complaints while remaining updated on the progress of their requests. Admins would then process these queries and provide feedback from FYP and finance departments along with any approved documents.

## 1.5 Product Scope

The Degree Issuance Module is a critical component of the One Stop Services Centre system at FAST Islamabad campus. It aims to digitize and automate the degree issuance process to address the challenges posed by the current manual and paper-based system. The primary objectives of the Degree Issuance Module include:

- Facilitating timely graduation of students
- Ensuring accuracy in academic records
- Improving operational efficiency for students and administration

The scope also extends to the implementation of robust security measures with a user-friendly and intuitive interface design.

## 1.6 References

1. Software Requirements Specification Document
  - Author: Ahmad Farhan
  - Version: 1.0
  - Date: 6<sup>th</sup> April 2024
  - Source: [Link](#)
2. User Stories and Acceptance Criteria Document
  - Author: Ahmad Farhan
  - Version: 1.0
  - Date: 5<sup>th</sup> March 2024
  - Source: [Link](#)

## 2. Overall Description

### 2.1 Product Perspective

The Degree Issuance System at FAST Islamabad campus is a standalone web-based application designed to streamline and automate the process of issuing degrees to students. It operates within the broader context of the One Stop Services Centre, which aims to provide comprehensive assistance to students at a single point of contact. While it interacts with various stakeholders and systems, its primary function is to facilitate the processing and issuance of degrees for undergraduate and graduate programs.

Interfaces and Dependencies:

- The system interfaces with the existing Flex and Slate systems at FAST Islamabad campus to retrieve student data and academic records.
- It interacts with the finance department's payment system to verify outstanding fees and process degree issuance fees.

### 2.2 Product Functions

Students:

- Submit Degree Issuance Form: Lets students submit requests for degree issuance.
- Submit Complaint Form: Lets students report errors in their academic documents.
- Make Payments: Allows students to pay degree issuance processing fees.
- Track Activity: Provides students with real-time updates on the status of their requests and notifications of any changes.
- Receive Degree: Students can receive digital copies of their degrees and notifications to collect physical copies.

One Stop Admin:

- Receive Notifications: Admins receive notifications of degree issuance requests and objections from the FYP and finance departments.
- Generate Tokens/Tickets: Generates unique tokens for each request with estimated processing timeframes.
- View Status of All Requests: Allows admins to view pending, processed, and new requests and manage objections.
- Issue Degrees: Admins issue degrees to students and update the request list.
- Issue Transcripts: Issues transcripts containing relevant academic information.



**Director:**

- View Daily Requests: Allows the director to monitor daily request activity.
- View Processing Metrics: Allows the director to view request processing metrics.
- Monitor Pending and Processed Requests: Provides visibility into pending and processed requests and department-wise processing times.
- Activity Tracking: Enables the director to track the status of each request.

**FYP Department:**

- Student Notification for Degree Issuance: Notifies students of updates in the processing of their forms.
- Provide Decision: Allows the FYP department to accept, reject, or raise objections to degree issuance requests and provide detailed comments.

**Finance Department:**

- Student Notification for Degree Issuance: Notifies students of updates in the processing of their forms.
- Verify All Payments Made: Verifies outstanding fees and degree issuance fees, allowing the finance department to accept, reject, or raise objections to requests.

## **2.3 User Classes and Characteristics**

The degree issuance system caters to the following user classes:

**Students:**

- Characteristics: Authority to make issuance requests and receive documentation.
- Responsibilities: Submit degree issuance requests, track status, receive notification and updates, receive degree and transcript documents.

**Onestop Admin:**

- Characteristics: Administrative privileges, oversight of degree issuance process, ability to manage objections and issue degrees.
- Responsibilities: Receive and manage degree issuance requests, resolve objections, issue degrees and transcripts.

**Director:**

- Characteristics: Executive role, responsible for oversight and decision-making.
- Responsibilities: Monitor request activity, track processing times, ensure efficient operation of the degree issuance system.

FYP Department:

- Characteristics: Technical expertise in academic evaluation, authority to approve or reject degree issuance requests.
- Responsibilities: Review and process degree issuance requests, provide feedback and decisions.

Finance Department:

- Characteristics: Financial expertise, responsible for verifying fee payments and financial clearance for degree issuance.
- Responsibilities: Verify payments, review, and process degree issuance requests, provide feedback and decisions.

## **2.4 Operating Environment**

The degree issuance system operates in a web-based environment, accessible through standard web browsers such as

- Google Chrome
- Mozilla Firefox
- Microsoft Edge
- Opera GX

It requires a stable internet connection for real-time communication with users and external services. Additionally, the system relies on server-side infrastructure to store and process data securely.

## **2.5 Design and Implementation Constraints**

Stack: C# with ASP.Net Framework and MS SQL Server Database.

Browser: Chrome, Edge, Mozilla Firefox, and Opera GX.

- Security: The system must adhere to security protocols and standards to protect sensitive student data and prevent unauthorized access.
- Scalability: The system should be designed to accommodate potential future growth in user volume and functionality.
- Integration: The system must seamlessly integrate with existing systems such as the Flex, Slate, and finance department payment system.

## **2.6 User Documentation**

Comprehensive user documentation will be provided for all user classes, including students, FYP and finance department staff, One Stop admins, and the director. The documentation will include:

- User Guides: Step-by-step instructions on how to use the system's features and functionalities.
- FAQs: Frequently asked questions and troubleshooting tips to address common issues.
- Contact Information: Contact details for technical support and assistance.

## **2.7 Assumptions and Dependencies**

Assumptions:

- The system assumes that users have basic computer literacy skills and internet access.
- It assumes that students provide accurate and complete information when submitting degree issuance requests.
- The system assumes that the existing IT infrastructure at FAST Islamabad campus can support the implementation and operation of the degree issuance system.

Dependencies:

- The system depends on access to the student information system used by Flex and slate to retrieve student data and academic records.
- It depends on the finance department's payment system to verify fee payments and financial clearance.

## 3. External Interface Requirements

### 3.1 User Interfaces

The user interface of the Degree Issuance System will be a web-based application accessible through standard web browsers. It will feature an intuitive and user-friendly design to facilitate ease of use for students, administrators, and other stakeholders. Key user interface elements will include:

- 1.1. Dashboard: Upon logging in, users will be greeted with a dashboard displaying relevant information and quick access to essential user specific functions.
- 1.2. Degree Issuance Form: Students shall interact with a form to submit degree issuance requests, with fields for personal information, program details, and any necessary documentation.
- 1.3. Complaint Form: A separate complaints form shall allow users to submit complaints regarding errors in academic documents.
- 1.4. Notifications: The system shall provide notifications to users regarding updates on their requests, ensuring they remain informed throughout the process.
- 1.5. Request Management: Admins and director shall have a dedicated interface for viewing and managing all degree issuance requests, including filtering options and status updates.
- 1.6. Request Evaluation: FYP and Finance departments shall have a dedicated interface for evaluating requests.
- 1.7. Request Processing Metrics: Director shall have an interface detailing the request processing metrics such as department-wise processing times.
- 1.8. Degree and Transcript Issuance: Students shall have an interface for receiving digital copies of their degree and transcript documents.

### 3.2 Hardware Interfaces

The Degree Issuance System will operate on standard hardware infrastructure commonly used for web-based applications. It will require:

- 3.1. Servers: System shall be hosted on an Apache Web server to store data securely
- 3.2. Client Devices: Users shall access the system via desktop computers or laptops with internet connectivity.

### 3.3 Software Interfaces

The system will integrate with various software components to support its functionality:

- 5.1. Database Management System: System shall interact with DBMS to store and manage student information, issuance requests, and administrative data.
- 5.2. Web Server: System shall use an Apache Web to facilitate communication between the client-side interface and the database of the web application.
- 5.3. Payment Gateway: System shall be Integrated with the payment processing system of FAST Islamabad to verify fee payments related to degree issuance.
- 5.4. Notification Services: System shall use email and SMS services to deliver notifications to users.

### 3.4 Communications Interfaces

The system will utilize standard communication protocols to facilitate interactions between different components:

- 6.1. HTTPS: System shall use HTTPS for communication between the web server and client devices to ensure secure data transmission over the internet.
- 6.2. SMTPS: System shall use SMTPS to send email notifications to users regarding updates on their degree issuance requests.
- 6.3. SMS: System shall send message notifications to users through SMS Gateways.

These interfaces will ensure seamless communication and interaction between users, administrators, and the underlying system components, facilitating efficient processing and management of degree issuance requests.

## 4. System Features

### 4.1 Degree Issuance Request Submission

**User Story: S-01**

As a: student  
I want: to submit a degree issuance form,  
So that: I can initiate an efficient process of obtaining my degree.

Importance:

5

Estimate:

5

**Acceptance Criteria:**

1. Given that: I want to initiate a degree issuance request  
When: I navigate to the degree issuance page  
Then: I should be able to fill out and submit a form

2. Given that I am on the degree issuance section  
When: I fill out the form with personal information  
program details and any relevant documentations,  
Then: I should be able to submit the form

Type:

- ☐ Search
- ☒ Workflow
- ☐ Manage Data
- ☐ Payment
- ☐ Report/View

**Functional Requirements:**

- 1.1 System shall allow students to make a degree issuance request.
  - 1.1.1 System shall provide students with a web form where they can input their details.
  - 1.1.2 The form should include fields for personal information (name, father name, roll number) and program details(department, batch no.).
  - 1.1.3 The user should be able to submit the filled-out degree issuance form.
- 1.2 System shall only accept completely and correctly filled forms.
  - 1.2.1 Upon submission, the system shall validate the data entered by the user in each field of the degree issuance form for completeness and correctness.
  - 1.2.2 If any required fields contain invalid data, the system shall display an error message.
  - 1.2.3 After successful submission of the form, the system shall display a confirmation message.
- 1.3 System must maintain a record of all the details filled in degree issuance forms.

## 4.2 Complaint Form Submission

**User Story: S-02**

As a: student  
I want: to submit complaint forms,  
So that: the university can look at and resolve any  
discrepancies or errors in my issued degree promptly.

Importance:

4

Estimate:

4

**Acceptance Criteria:**

1. Given that: I have received an academic document  
When: I identify any discrepancies or errors  
Then: I should be able to submit a complaint form  
to report the issue

2. Given that: I submit a complaint form  
When: the university receives my complaint  
Then: they should acknowledge and resolve my query

Type:

- ☐ Search
- ☒ Workflow
- ☐ Manage Data
- ☐ Payment
- ☐ Report/View

**Functional Requirements:**

- 1.1 System shall allow students to submit a complaint form.
  - 1.1.1 System shall provide students with a complaints section for submitting complaint forms.
  - 1.1.2 System shall provide a complaint form that includes fields for identifying discrepancies (spelling mistakes etc.) in the degree or transcript document received.
- 1.2 System shall acknowledge the receipt of the complaint form.
  - 1.2.1 Upon submission, the system shall validate the data entered by the user in each field of the complaint form for completeness and correctness.
  - 1.2.2 If any required fields contain invalid data, the system shall display an error message.
  - 1.2.3 After successful submission of the form, the system shall display a confirmation message.
- 1.3 System must maintain a record of all the details filled in complaint forms.

### 4.3 Query Progress Tracking

#### User Story: S-03

As a: student  
I want: to track the real-time progress of my queries,  
So that: I can monitor the progress and know the expected timeline for receiving my degree.

Importance:

3

Estimate:

3

#### Acceptance Criteria:

1. Given that: I have submitted the degree issuance form  
When: I review the confirmation message  
Then: it should indicate that my request has been received and is being processed.
2. Given that: I submit any a degree issuance request form  
When: the university begins processing my request  
Then: I should see the status of my query in real-time.

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

#### Functional Requirements:

- 2.1 System shall provide users with real-time progress updates on the status of their complaints and degree issuance requests.
- 2.2 System shall inform the student about the current status with the timestamp of last update for their request.

### 4.4 Receipt of Digital Documents

#### User Story: S-04

As a: student  
I want: to receive digital copies of my degree and transcripts,  
So that: I can use them as needed.

Importance:

5

Estimate:

2

#### Acceptance Criteria:

Given that: I have submitted a degree issuance request,  
When: I meet all the requirements and the issuance process has been completed  
Then: I should be issued a copy of my transcript and degree

Type:

- ☐ Search
- ☒ Workflow
- ☐ Manage Data
- ☐ Payment
- ☐ Report/View



**Functional Requirements:**

- 3.1 Upon completion of the degree issuance process, the system shall automatically issue a digital copy of the student's degree and transcript to their account.

**4.5 Notifications for Degree Collection****User Story: S-05**

As a: student  
I want: to be notified when my degree is ready for collection,  
So that: I can arrange to receive the physical certificate from the university.

Importance:

4

Estimate:

2

**Acceptance Criteria:**

Given that: I have submitted a degree issuance request,  
When: the university has processed my issuance request  
Then: I should be notified that my degree is ready for collection

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

**Functional Requirements:**

- 4.1 Upon completion of the degree issuance process, the system shall automatically notify the student that their degree is ready for collection.
- 4.2 System will notify the student within 24-hours about the successful completion of their degree issuance request.

**4.6 Student Account Registration****User Story: S-06**

As a: student  
I want: to sign into the system with my flex credentials,  
So that: I do not have to remember the credentials for yet another university system.

Importance:

3

Estimate:

1

**Acceptance Criteria:**

Given that: I am a student at the university,  
When: I want to log into the system  
Then: I should not have to sign up explicitly and should be able to log in using my flex credentials.

Type:

- ☐ Search
- ☐ Workflow
- ☒ Manage Data
- ☐ Payment
- ☐ Report/View

**Functional Requirements:**

- 5.1 System will allow students to login using their credentials from their flex account.

**4.7 Payment of Degree Issuance Request Fees****User Story: S-07**

As a: student  
 I want: pay my degree issuance request fees,  
 So that: the university may begin processing my request and issue my degree without any complications.

Importance:

5

Estimate:

4

**Acceptance Criteria:**

1. Given that: I am a student with an active account,  
 When I submit a degree issuance request,  
 Then: I should be able to pay for my issuance request.
2. Given that: I have entered the payment gateway,  
 When: I input my payment details and confirm the payment,  
 Then: I should receive a confirmation message indicating the success of the payment.

Type:

- ☐ Search  
☐ Workflow  
☐ Manage Data  
☒ Payment  
☐ Report/View

**Functional Requirements:**

- 6.1 System must provide a secure payment gateway where students can submit their fees electronically.
- 6.2 Students should receive confirmation after the successful payment of fees.
- 6.3 System should record the payment details for future reference and verification.
- 6.4 The university staff should be able to verify the payment status.

**4.8 Status Updates Notification****User Story: S-08**

As a: student  
 I want: to receive notifications or updates via SMS or email whenever there is any change in the status of my queries,  
 So that: I can stay updated without having to check the system constantly.

Importance:

3

Estimate:

3

**Acceptance Criteria:**

Given that: I submit any degree issuance request form,  
 When: the university is processing my request,  
 Then: I should receive updates about the progress of my query.

**Type:**

- ☐ Search  
☐ Workflow  
☐ Manage Data  
☐ Payment  
☒ Report/View

**Functional Requirements:**

- 7.1 When a user's degree issuance request is being processed, the system shall automatically send notifications to the user about the progress of their query via SMS or email.

**4.9 Degree Issuance Request Notification - Admin****User Story: A-01**

As a: One stop Admin  
 I want: to be notified when a student submits a degree issuance request,  
 So that: I can promptly begin processing their queries.

**Importance:**

4

**Estimate:**

3

**Acceptance Criteria:**

Given that: I am logged into the system as an admin,  
 When: a student submits a degree issuance request,  
 Then: I should be given notifications.

**Type:**

- ☐ Search  
☐ Workflow  
☐ Manage Data  
☐ Payment  
☒ Report/View

**Functional Requirements:**

- 8.1 System should notify user when a degree issuance request has been submitted.  
 8.1.1 System should have a section dedicated to showing request notifications.  
 8.1.2 System should have indicators on notifications distinguishing new notifications.  
 8.1.3 Notifications shown should have timestamp of request submission time.  
 8.2 In the notification, the system shall provide quick access to relevant details of the submission.

## 4.10 Token Generation with Time Estimates

### User Story: A-02

As a: One stop Admin  
 I want: to generate unique tokens with estimated timeframes for every student request,  
 So that: they can be processed on a first come first served basis.

Importance:

5

Estimate:

5

### Acceptance Criteria:

1. Given that: a student submits a request  
 When: the request is received  
 Then: a unique token should be generated and assigned to the request.
2. Given that: a student submits a request  
 When: the token has been generated  
 Then: the student should be notified with an estimated timeframe of processing the request.
3. Given that: tokens are generated for student requests  
 When: multiple requests are received  
 Then: the tokens should be assigned on a first come first served basis.

Type:

- ☐ Search
- ☐ Workflow
- ☒ Manage Data
- ☐ Payment
- ☐ Report/View

### Functional Requirements:

- 9.1 When a degree issuance request is submitted, system should automatically generate a unique token for the request.
  - 9.1.1 System will calculate the estimated time of completion of request.
  - 9.1.2 User should be notified with the estimated time of completion.
- 9.2 The system will assign the generated unique token to the new request.
  - 9.2.1 System will assign tokens on a first come first served basis.

## 4.11 Degree Issuance Upon Approval

### User Story: A-03

As a: One stop Admin  
 I want: to issue degrees to the students upon approvals from all the relevant authorities,  
 So that: the students can be informed of the completion of their requests and receive their degrees.

Importance:

5

Estimate:

4

**Acceptance Criteria:**

Given that: a degree issuance process is initiated,  
 When: the request has been approved by all the relevant authorities  
 Then: the degree should be issued to the student.

**Type:**

- ☐ Search  
☒ Workflow  
☐ Manage Data  
☐ Payment  
☐ Report/View

**Functional Requirements:**

- 10.1 Onestop admins should be able to issue degrees to students.
- 10.1.1 When a request is given approval from both FYP and Finance departments, the system shall finalize the degree issuance request.
- 10.1.2 When a request is finalized, the student will be issued their degree document.

**4.12 Request Management****User Story: A-04**

As a: One stop Admin  
 I want: to view the status of all pending, processed, and new requests  
 So that: I can manage and prioritize them effectively.

**Importance:**

4

**Estimate:**

3

**Acceptance Criteria:**

1. Given that: I am logged into the system,  
 When: I navigate to request management section  
 Then: I should see a list of all pending, processed and new requests along with their relevant details like student names and dates.
2. Given that: I am viewing the list of students' requests,  
 When: I need to filter or sort the list  
 Then: I should be able to sort by status or submission date

**Type:**

- ☐ Search  
☐ Workflow  
☐ Manage Data  
☐ Payment  
☒ Report/View

**Functional Requirements:**

- 11.1 User should be able to view the status of all pending, processed, and new requests.
- 11.2 System should provide admins with a dedicated request management section.
- 11.3 The requests in the management section should have relevant details such as student names, and dates of submission.
- 11.4 System shall let users filter the list by submission date and status.
- 11.5 System shall let users sort the list by submission date and status.

## 4.13 Transcript Generation

### User Story: A-05

As a: One stop Admin  
 I want: the transcripts to automatically be generated for students when their degree issuance request is approved,  
 So that: the students can receive their academic documentation.

Importance:

5

Estimate:

3

### Acceptance Criteria:

1. Given that: the degree issuance process is complete,  
 When: the student is issued their degree,  
 Then: their transcript should also be issued.
2. Given that: a student requests their transcript,  
 When: the transcript is generated and provided to the student,  
 Then: the transcript should include details such as department, CGPA, course grades, duration and photo.

Type:

- ☐ Search
- ☒ Workflow
- ☐ Manage Data
- ☐ Payment
- ☐ Report/View

### Functional Requirements:

- 12.1 User should be able to receive their transcripts from the system.
- 12.2 When a degree issuance request is finalized, the system shall automatically generate student transcripts.
- 12.3 When a degree issuance request is finalized, the system shall provide the student with their transcripts.
- 12.4 The Transcripts should contain details such as department, CGPA, course grades, and duration.

## 4.14 Handling Request Objections

### User Story: A-06

As a: One stop Admin  
 I want: to know about any objections raised by the finance or FYP departments regarding a student's degree issuance,  
 So that: I can inform the student of the issue and guide them about the next steps.

Importance:

5

Estimate:

5

**Acceptance Criteria:**

1. Given that a degree issuance request is under review, when objections are raised by the FYP or finance departments then I should be informed about the objections
2. Given that objections are raised against a student request, when I have reviewed the objections then I should be able to notify the student
3. Given that objections are raised against a student request, when I am notifying the student then I should be able to include the details of reason for objection and guide them about the next step.

**Type:**

- ☐ Search  
☒ Workflow  
☐ Manage Data  
☐ Payment  
☐ Report/View

**Functional Requirements:**

- 13.1 System should inform the admin about all objections raised by the FYP and finance departments, against a request.
- 13.2 Admin should be able to inform students about the objections raised against their requests.
- 13.3 Admin should be able to inform the student with the reasons for objection and instructions about what to do next using comments.

**4.15 Automatic List Updates****User Story: A-07**

As a: One stop Admin  
 I want: the requests list to automatically be updated when a degree is issued,  
 So that: it is clear which requests have been processed.

**Importance:**

4

**Estimate:**

3

**Acceptance Criteria:**

Given that: a degree has been issued to the student,  
 When: the degree issuance process is complete  
 Then: the requests list should automatically be updated.

**Type:**

- ☐ Search  
☐ Workflow  
☒ Manage Data  
☐ Payment  
☐ Report/View

**Functional Requirements:**

- 14.1 When a degree is issued, the system shall automatically update the request status in the requests list.

## 4.16 Filter Requests by Day

### User Story: D-01

As a: Director  
 I want: to view a list of all the requests generated on a specific day,  
 So that: I have a comprehensive overview of daily activities.

Importance:

3

Estimate:

3

### Acceptance Criteria:

1. Given that: I am viewing the list of student requests,  
 When: I need to view all requests received on a specific day  
 Then: I should be able to filter the requests by date.
2. Given that: I am logged in as the director,  
 When: I filter the requests by current date  
 Then: I should see a list of all requests generated on a specific day.

Type:

- ☒ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☐ Report/View

### Functional Requirements:

- 15.1 The director should be able to filter degree issuance requests by specific days to track daily activities and trends.

## 4.17 View Pending Requests

### User Story: D-02

As a: Director  
 I want: to access a list of all pending requests,  
 So that: I can monitor the remaining workload and identify bottlenecks in processing.

Importance:

3

Estimate:

2

### Acceptance Criteria:

Given that: I am viewing the list of student requests,  
 When: I filter the list to view all pending requests  
 Then: I should see all pending requests with details list student name and submission date.

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

### Functional Requirements:

- 16.1 The director should have the capability to view a list of all pending degree issuance requests to monitor remaining workload and identify processing bottlenecks.



## 4.18 View Processed Requests

**User Story: D-03**

As a: Director  
I want: to view all the processed requests,  
So that: I can track the efficiency of the process.

Importance:

2

Estimate:

2

**Acceptance Criteria:**

Given that: I am viewing the list of student requests,  
When: I filter the list to view all processed requests  
Then: I should see all processed requests with details  
list student name and finalization date.

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

**Functional Requirements:**

17.1 The Director should have access to detailed information about each degree issuance request for oversight and decision-making purposes.

## 4.19 View Processing Metrics

**User Story: D-04**

As a: Director  
I want: to view processing details such as the time taken by  
each department to process a request,  
So that: I can identify bottlenecks and optimize the processes  
accordingly.

Importance:

4

Estimate:

4

**Acceptance Criteria:**

Given that: I am logged in as the director,  
When: I view the request processing metrics  
Then: I should see the time taken by each department to  
process the request

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

**Functional Requirements:**

18.1 The director should have access to processing details such as the time taken by each department to process requests, to identify bottlenecks and optimize processes.

## 4.20 Active Progress Tracking - Director

### User Story: D-05

As a: Director  
I want: to track the real-time progress of requests,  
So that: I can stay informed about the progress and status of every request.

Importance:

3

Estimate:

3

### Acceptance Criteria:

Given that: students submit any request form  
When: the university begins processing the request  
Then: I should be able to view the real-time activity log of each request.

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

### Functional Requirements:

19.1 The Director should be able to monitor the overall status of degree issuance requests, including pending, processed, and completed requests.

## 4.21 Request Processing Timestamps

### User Story: Y-01

As a: FYP Department Staff  
I want: each request to be time stamped when the processing begins and ends,  
So that: accurate record of processing times is maintained.

Importance:

3

Estimate:

4

### Acceptance Criteria:

1. Given that: a student submits a degree issuance request,  
When: the request is received, and I begin processing it  
Then: the request should automatically be time stamped with the date and time indicating the start of processing.
2. Given that: a student's request is under processing,  
When: I complete the processing the degree issuance request  
Then: the request should automatically be time stamped with the date and time indicating the completion of processing.

Type:

- ☐ Search
- ☐ Workflow
- ☒ Manage Data
- ☐ Payment
- ☐ Report/View

**Functional Requirements:**

- 20.1 When a request is initiated, the system should automatically record the timestamp (start time) for that request.
- 20.2 When the processing of the request is completed, the system should record the timestamp (end time).
- 20.3 The system should calculate and display the duration (processing time) between the start and end timestamps.

**4.22 Degree Issuance Request Notification - FYP****User Story: Y-02**

As a: FYP Department Staff  
 I want: to receive notifications when a student submits a degree issuance request,  
 So that: I can promptly begin processing their queries.

Importance:

3

Estimate:

3

**Acceptance Criteria:**

Given that: I am logged into the system,  
 When: a student submits a degree issuance request  
 Then: I should be given timely notifications

Type:

- ☐ Search  
☐ Workflow  
☐ Manage Data  
☐ Payment  
☒ Report/View

**Functional Requirements:**

- 21.1 The FYP Department staff should receive timely notifications when students submit degree issuance requests so that they can promptly begin processing them.

**4.23 Request Evaluation and Decision – FYP\****User Story: Y-03**

As a: FYP Department Staff  
 I want: to accept, reject, or raise objections to a degree issuance request,  
 So that: I can effectively process requests and inform the student of the decision.

Importance:

5

Estimate:

4

**Acceptance Criteria:**

1. Given that: a degree issuance request is under review,  
When: all required criteria are met by the student  
Then: I should be able to approve the issuance request
2. Given that: a degree issuance request is under review,  
When: I have any objections against the request  
Then: I should be able to provide detailed comments with guidance for next steps

**Type:**

- ☐ Search  
☒ Workflow  
☐ Manage Data  
☐ Payment  
☐ Report/View

**Functional Requirements:**

- 22.1 The FYP and Finance Department staff should be able to review and evaluate degree issuance requests submitted by students.
- 22.2 The FYP and Finance Department staff should be able to raise objections to degree issuance requests if they identify any discrepancies or issues that need clarification or further verification.
- 22.3 The FYP and Finance Department staff should have the capability to provide detailed feedback, including acceptance, rejection, or objections with comments, for each degree issuance request.

**4.24 Verification of Outstanding Fees\****User Story: F-01**

As a: Finance Department Staff  
 I want: to verify that all outstanding fees of a student including the issuance fees are paid,  
 So that: I can ensure that an issuance request meets the financial criteria for approval.

**Importance:**

5

**Estimate:**

5

**Acceptance Criteria:**

Given that: a degree issuance request is under review,  
 When: all financial criteria are met by the student  
 Then: I should be able to approve the issuance request

**Type:**

- ☐ Search  
☐ Workflow  
☐ Manage Data  
☒ Payment  
☐ Report/View

**Functional Requirements:**

- 23.1 Finance Department staff should be able to verify that all outstanding fees, including the degree issuance fee, have been paid by students before approving degree issuance requests.

## 4.25 Requests Evaluation and Decision - Finance

### User Story: F-02

As a: Finance Department Staff  
 I want: to accept, reject, or raise objections to a degree issuance request,  
 So that: I can effectively process requests and inform the student of the decision.

Importance:

5

Estimate:

4

### Acceptance Criteria:

1. Given that: a degree issuance request is under review,  
 When: all required financial criteria are met by the student  
 Then: I should be able to approve the issuance request
2. Given that: a degree issuance request is under review,  
 When: I have any objections against the request  
 Then: I should be able to provide comments for next steps

Type:

- ☐ Search
- ☒ Workflow
- ☐ Manage Data
- ☐ Payment
- ☐ Report/View

### Functional Requirements:

- 24.1 The FYP and Finance Department staff should be able to access detailed information about each degree issuance request, including student details, program information, and any attached pdf documents.

## 4.26 Degree Issuance Request Notification – Finance

### User Story: F-03

As a: Finance Department Staff  
 I want: to receive notifications when a student submits a degree issuance request,  
 So that: I can promptly begin processing their queries.

Importance:

3

Estimate:

3

### Acceptance Criteria:

Given that: I am logged into the system,  
 When: a student submits a degree issuance request  
 Then: I should be given timely notifications

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

### Functional Requirements:

- 25.1 The Finance Department staff should receive timely notifications when students submit degree issuance requests so that they can promptly begin processing them.

## **5. Other Nonfunctional Requirements**

### **5.1 Performance Requirements**

- 1.1 Real-Time Updates: The system should provide real-time access to request status and processing metrics with a latency of no more than 5 seconds.
- 1.2 Response Time: The system should respond to user actions, such as form submissions and data retrievals, within 2 seconds under normal load conditions.
- 1.3 Scalability: The system should be capable of managing an increasing number of degree issuance requests efficiently without performance degradation.

### **5.2 Safety Requirements**

- 2.1 Data Integrity: The system must prevent unauthorized access, modification, or deletion to ensure the integrity of user data.
- 2.2 Backup and Recovery: Regular backups of system data should be performed, with a recovery plan in place to restore data in case of system failure.

### **5.3 Security Requirements**

- 3.1 Access Control: Access to degree issuance request details should be restricted to authorized FYP and Finance Department staff to maintain confidentiality and integrity.
- 3.2 User Authentication: Users must authenticate themselves using username and password, before accessing sensitive information or performing critical actions.

### **5.4 Software Quality Requirements**

- 4.1 Accessibility: The interface for monitoring degree issuance requests should be accessible from different devices and platforms to accommodate the Director's needs for remote oversight.
- 4.2 Maintainability: Code should be well-documented and structured to facilitate ease of maintenance and future enhancements by developers.
- 4.3 Reliability: The system should reliably update and display the status of degree issuance requests to provide accurate insights for decision-making.
- 4.4 Usability: The system should be intuitive and user-friendly, with clear navigation and informative error messages to assist users in completing tasks efficiently.

### **5.5 Business Rules**

- 5.1 Approval Processes: Degree issuance requests should undergo approval processes by relevant departments, such as the Finance and Faculty of Final Year Project (FYP), before being finalized and issued to students.
- 5.2 Accuracy: The system should accurately calculate and verify outstanding fees to ensure the financial integrity of degree issuance requests.
- 5.3 Audit-Trail: All financial transactions related to degree issuance requests should be logged and auditable to maintain transparency and accountability.

## 6. Design

### 6.1 Architecture

The system is designed using a combination of MVC (Model-View-Controller) and 3-tier architecture. This architecture was chosen for its ability to separate concerns, improve maintainability, and enhance scalability of the system.

#### MVC Architecture:

- **Model Layer:** Encapsulates business logic and data handling, ensuring data integrity and abstraction.
- **Views Layer:** Provides the user interface components, facilitating a clear separation of concerns between the presentation and application logic.
- **Controller Layer:** Acts as intermediary between the Model and View components, orchestrating user interactions and business operations.

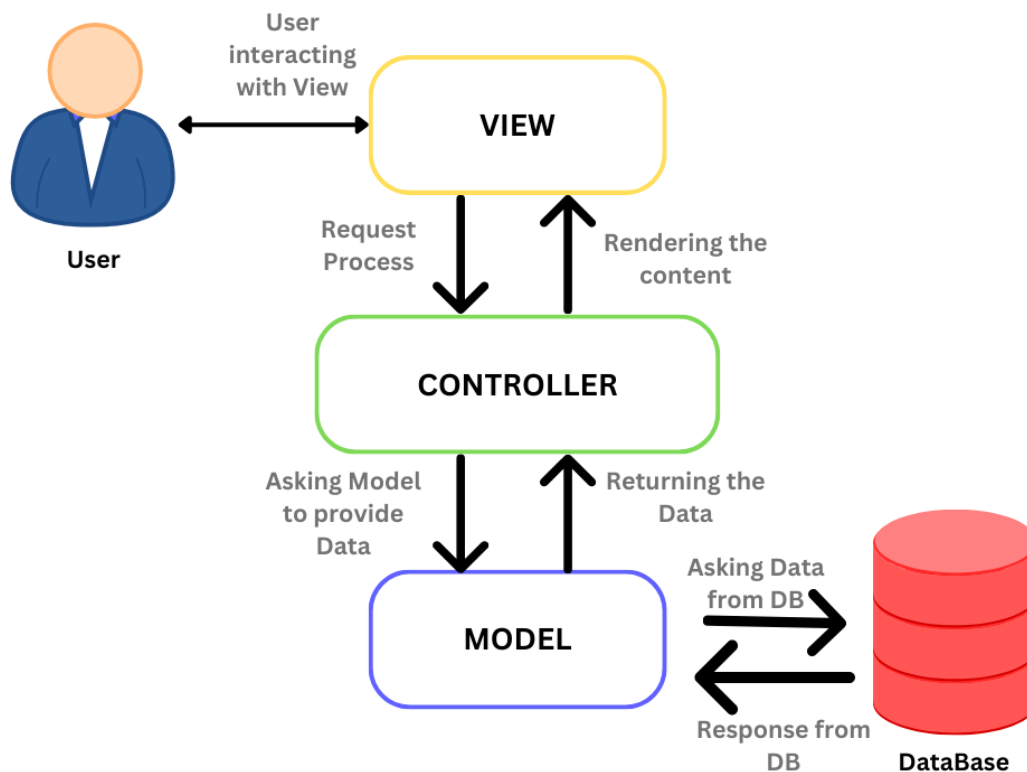
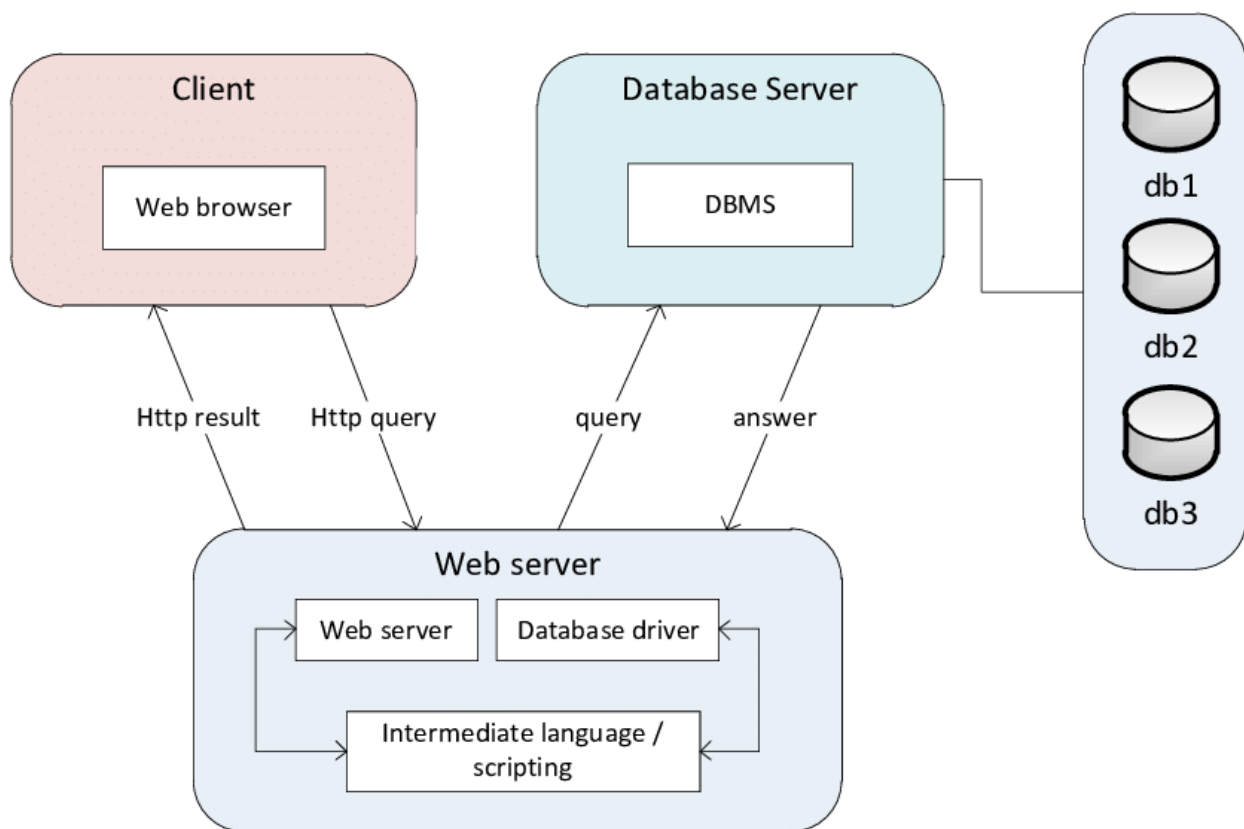


Diagram credits: [The MVC Architecture. MVC \(Model-View-Controller\) | by Sadika | Medium](#)



### 3-Tier Architecture:

- **Presentation Layer:** Contains all the classes responsible for presenting the UI to the end-user and handling browser communication logic. This is the only layer that customers interact with, ensuring data integrity and abstraction.
- **Business Logic Layer:** Contains all the logic and services required by the application to meet its functional requirements. This layer deals with data aggregation, computation and query requests, ensuring reusability and modularity.
- **Data Layer:** Manages data storage and access, ensuring reliability, and security.



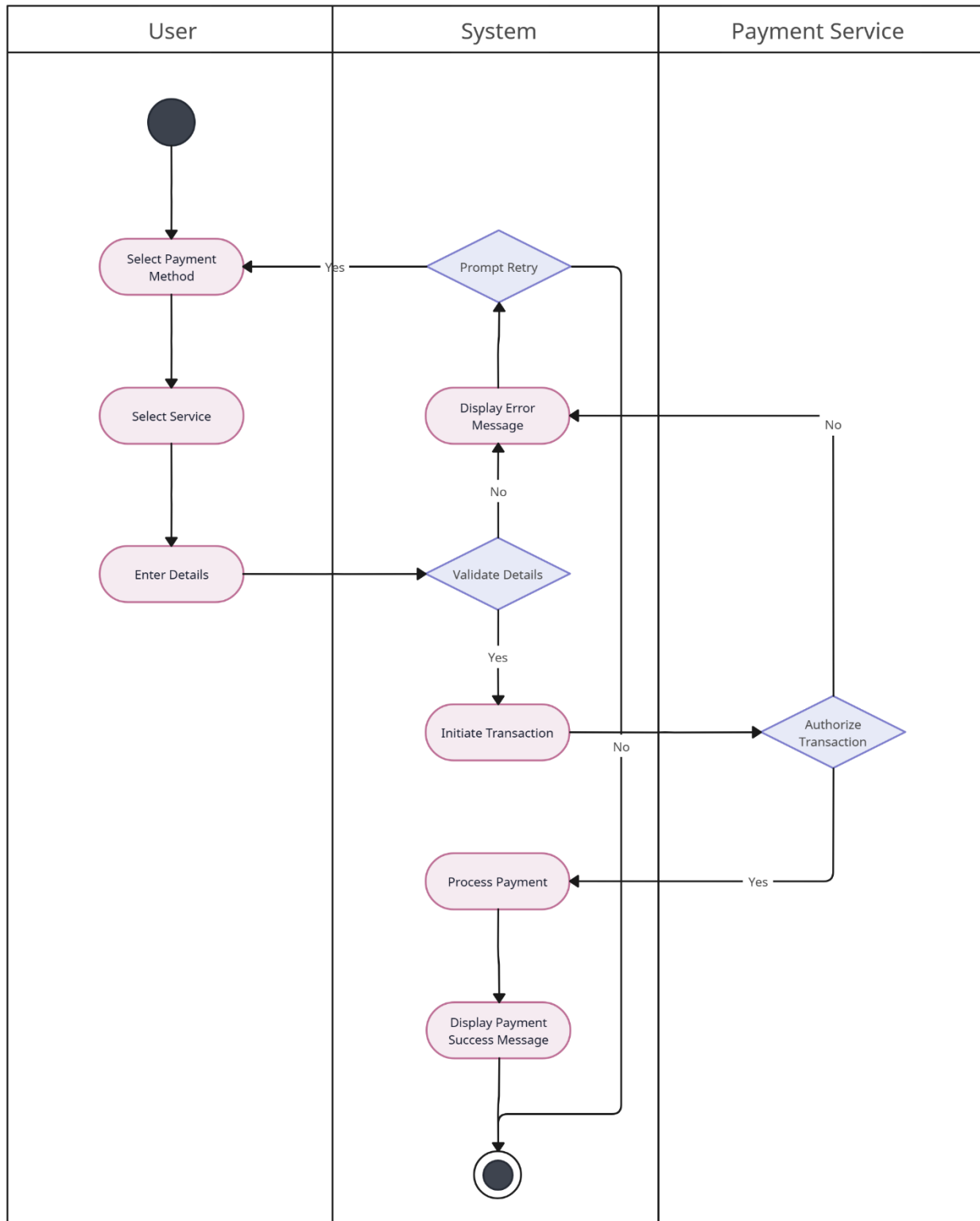
**Diagram Credits:** [The 3-tier architecture | Scientific Diagram \(researchgate.net\)](https://www.researchgate.net/publication/312111111)

## 6.2 Use Case Diagram

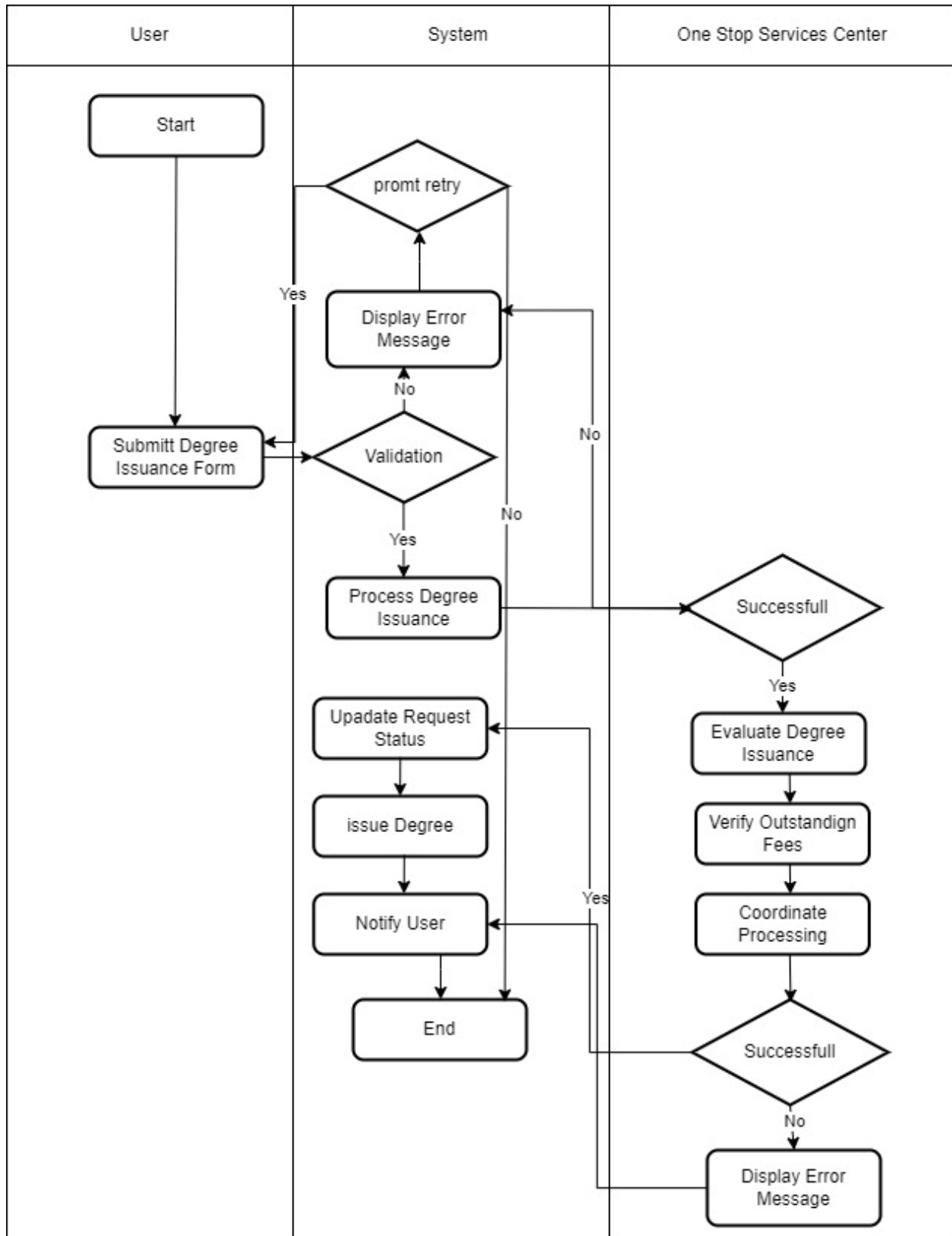


## 6.3 Activity Diagram

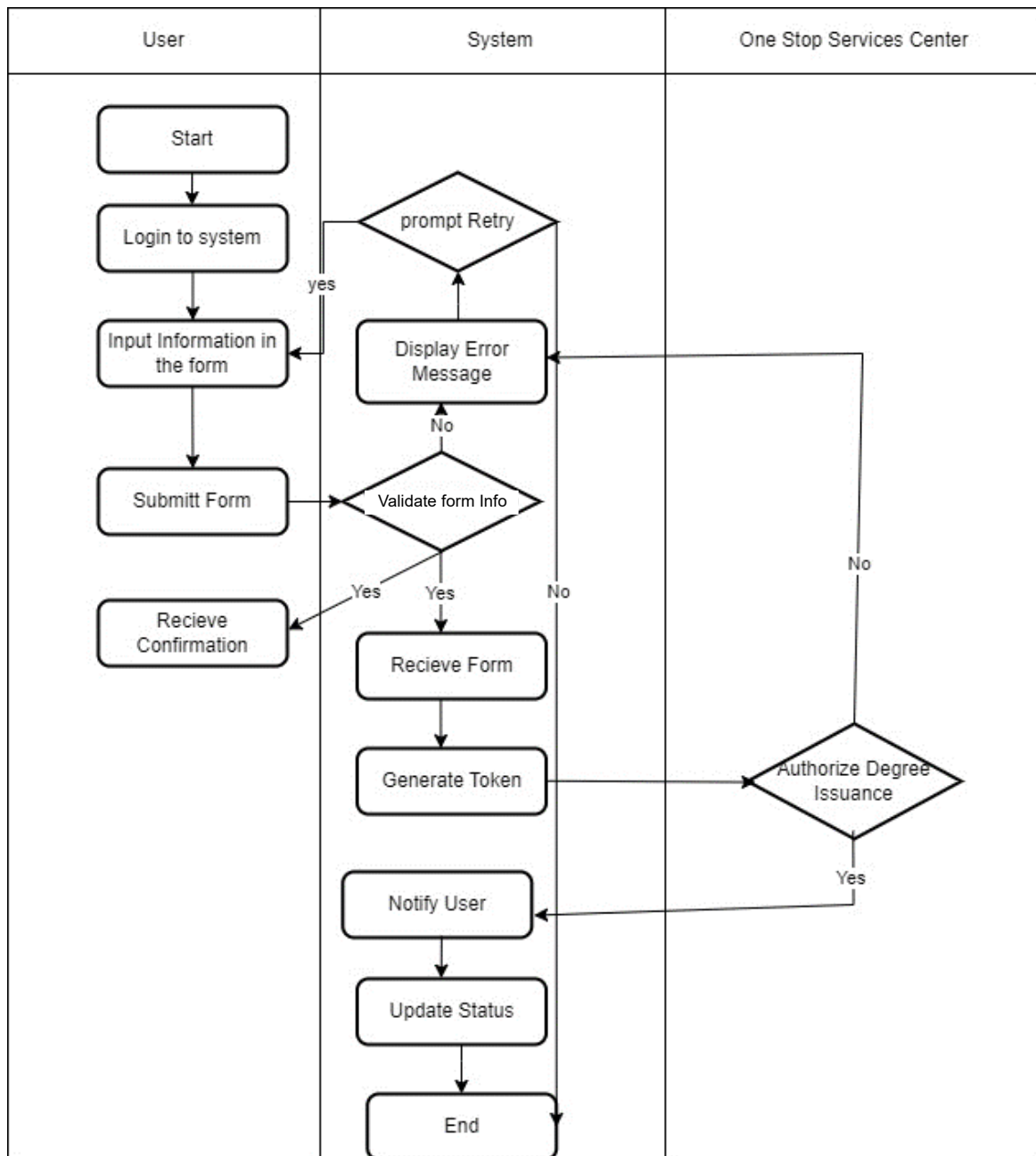
### 6.3.1 Make Payment



## 6.3.2 Receive Degree

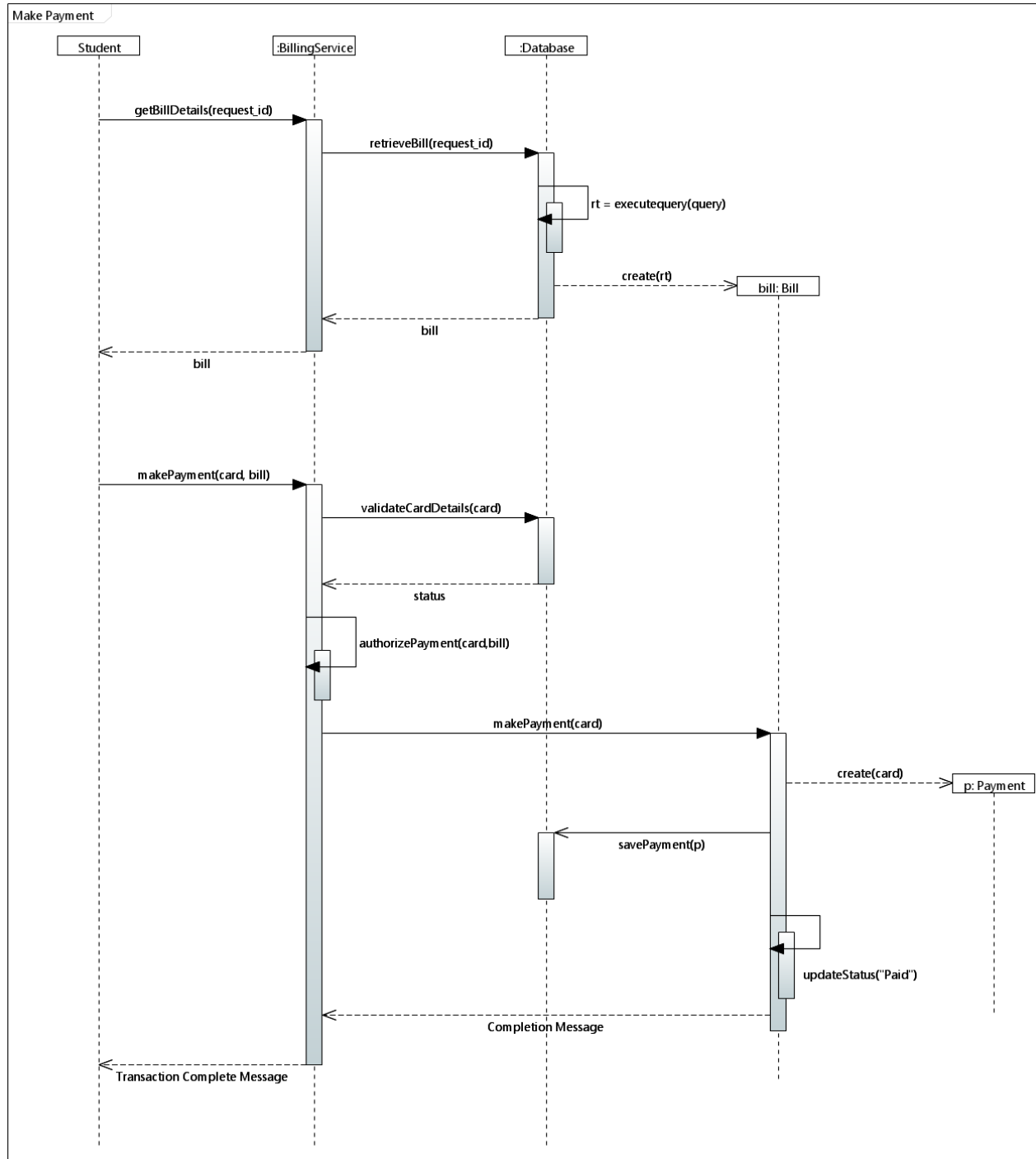


## 6.3.3 Initiate Degree Issuance Request

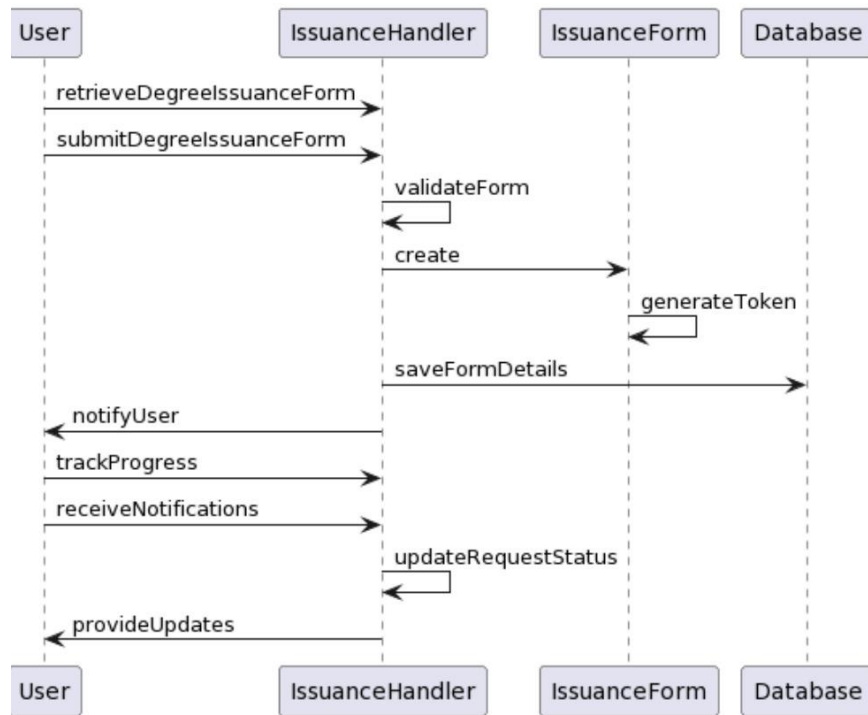


## 6.4 Sequence Diagram

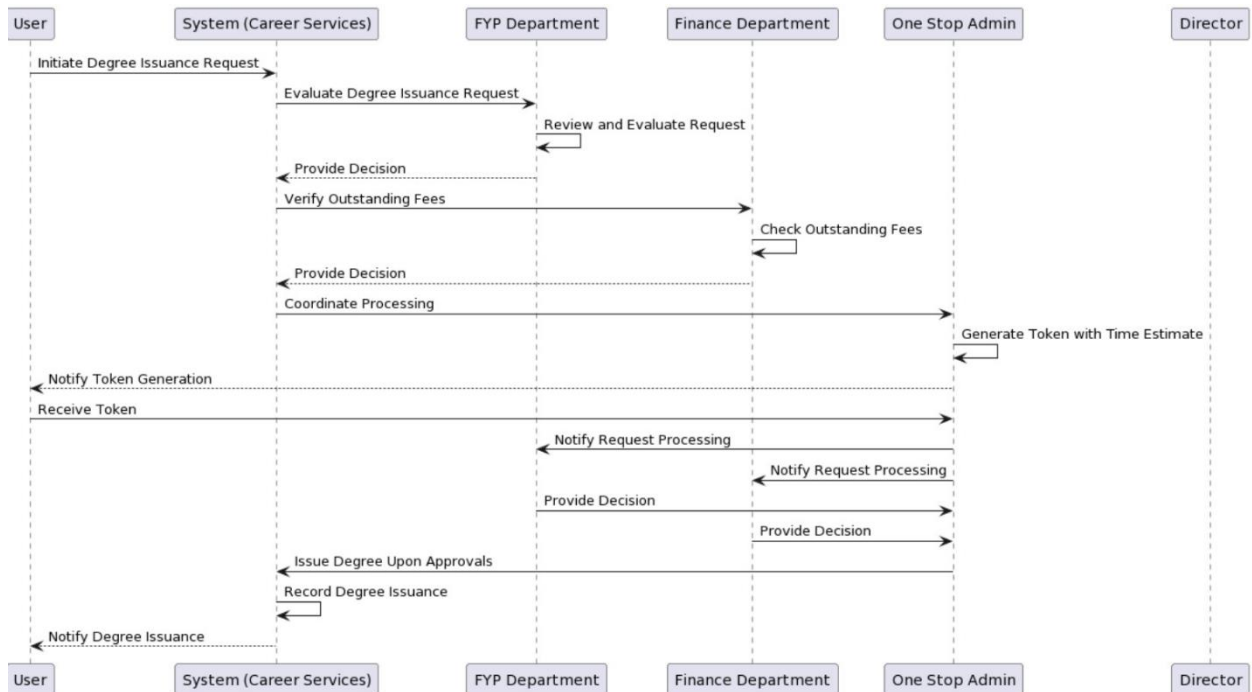
### 6.4.1 Make Payment



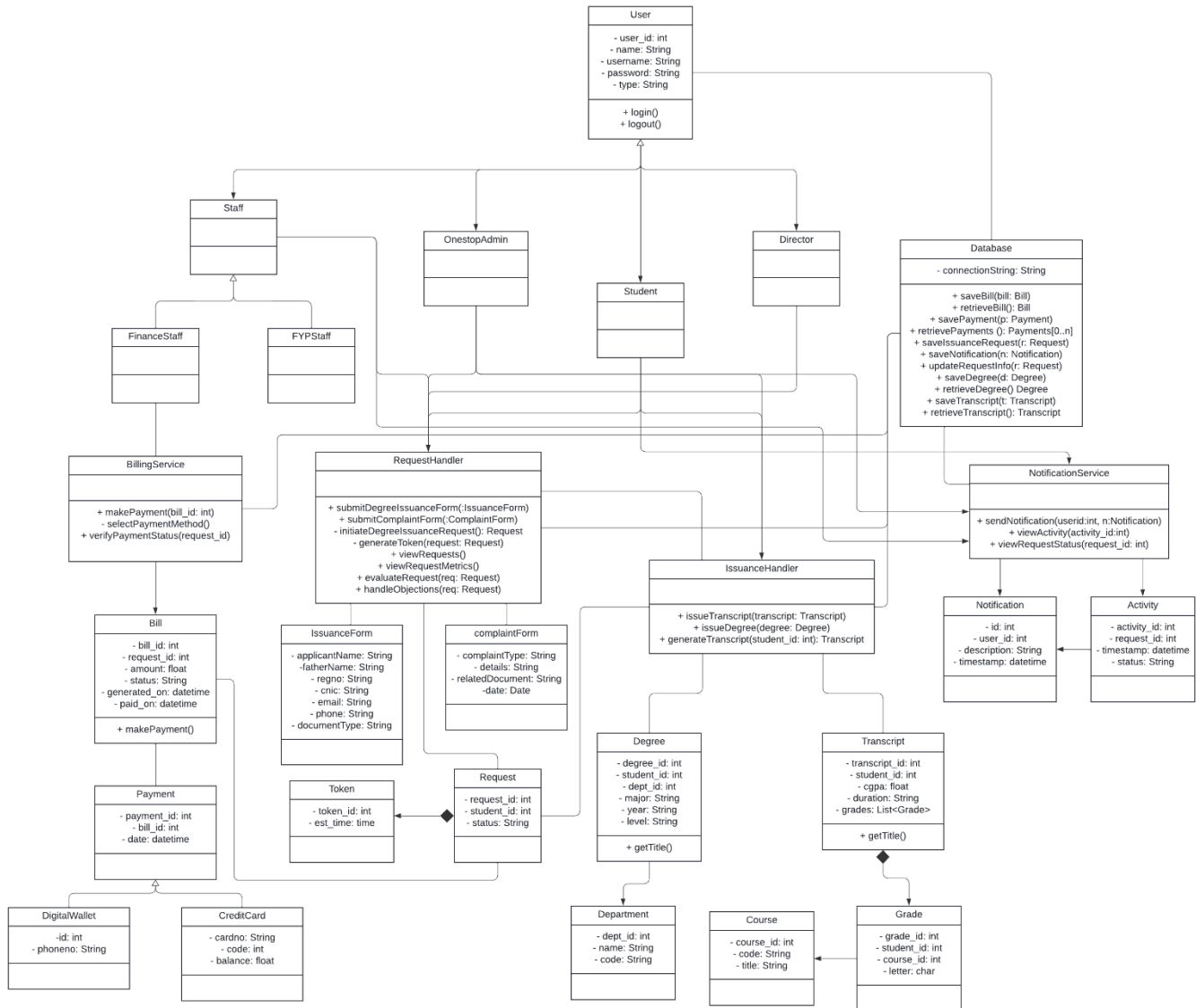
## 6.4.2 Initiate Degree Issuance Request



## 6.4.3 Receive Degree



## 6.5 Class Diagram





## 7. Product Backlog

### 7.1 User Story S-01

#### Degree Issuance Request Submission

As a: student

I want: to submit a degree issuance form,

So that: I can initiate an efficient process of obtaining my degree.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Develop web form for degree issuance request.
- Implement data validation for degree issuance form.
- Develop backend functionality for storing submitted form data.

#### Associated User Stories:

1. User Story S-06: Student Account Registration
2. User Story A-02: Token Generation with Time Estimates
3. User Story S-07: Payment of Degree Issuance Request Fees
4. User Story Y-02: Degree Issuance Request Notification - FYP
5. User Story A-01: Degree Issuance Request Notification - Admin
6. User Story Y-03: Degree Issuance Request Notification - Finance

### 7.2 User Story S-02

#### Complaint Form Submission

As a: student

I want: to submit complaint forms,

So that: the university can look at and resolve any discrepancies or errors in my issued degree promptly.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Develop web form for complaint submission.
- Implement data validation for complaint form fields.
- Develop backend functionality for storing submitted complaint data.

**Associated User Stories:**

1. User Story S-06: Student Account Registration
2. User Story S-01: Degree Issuance Request Submission

**7.3 User Story S-03****Query Progress Tracking**

As a: student

I want: to track the real-time progress of my queries,

So that: I can monitor the progress and know the expected timeline for receiving my degree.

**Context:**

This story is related to the sprint-02.

**Tasks:**

- Implement real-time progress updates.
- Implement backend functionality to update query progress in real-time.
- Integrate frontend UI with backend for displaying real-time progress updates.

**Associated User Stories:**

1. User Story S-01: Degree Issuance Request Submission
2. User Story S-08: Status Updates Notification

**7.4 User Story S-04****Receipt of Digital Documents**

As a: student

I want: to receive digital copies of my degree and transcripts,

So that: I can use them as needed.

**Context:**

This story is related to the sprint-03.

**Tasks:**

- Develop functionality to generate digital copies of transcripts and degrees.
- Implement verification mechanisms to ensure all requirements are met before issuing documents.
- Design user interface for viewing digital documents.

**Associated User Stories:**

1. User Story S-01: Degree Issuance Request Submission
2. User Story S-05: Notifications for Degree Collection
3. User Story A-05: Transcript Generation
4. User Story A-03: Degree Issuance Upon Approval

**7.5 User Story: S-05****Notifications for Degree Collection**

As a: student

I want: to be notified when my degree is ready for collection,

So that: I can arrange to receive the physical certificate from the university.

**Context:**

This story is related to the sprint-02.

**Tasks:**

- Develop mechanism to track degree issuance status and trigger notifications.
- Design user interface for viewing and managing notifications.

**Associated User Stories:**

1. User Story S-01: Degree Issuance Request Submission
2. User Story A-05: Transcript Generation
3. User Story A-03: Degree Issuance Upon Approval

**7.6 User Story: S-06****Student Account Registration**

As a: student

I want: to sign into the system with my flex credentials,

So that: I do not have to remember the credentials for yet another university system.

**Context:**

This story is related to the sprint-01.

**Tasks:**

- Integrate Flex authentication system with university's login mechanism.
- Develop user interface for students to sign in using Flex credentials.
- Implement backend logic to validate Flex credentials and authenticate users.

## 7.7 User Story: S-07

### Payment of Degree Issuance Request Fees

As a: student

I want: to pay my degree issuance request fees,

So that: the university may begin processing my request and issue my degree without any complications.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Implement a secure payment gateway for students to submit fees electronically.
- Develop user interface for students to view and pay degree issuance request fees.
- Integrate payment confirmation mechanism with degree issuance request.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.8 User Story: S-08

### Status Updates Notification

As a: student

I want: to receive notifications or updates via SMS or email whenever there is any change in the status of my queries,

So that: I can stay updated without having to check the system constantly.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Set up a notification system to send updates via SMS or email.
- Define triggers for status changes that will trigger notifications.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission
2. User Story A-04: Request Management

## 7.9 User Story: A-01

### Degree Issuance Request Notification - Admin

As a: One stop Admin

I want: to be notified when a student submits a degree issuance request,

So that: I can promptly begin processing their queries.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Send notification to admin when a degree issuance request is submitted.
- Configure notification triggers to notify admin upon request submission.
- Develop user interface for admin to manage notification preferences.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission
2. User Story A-04: Request Management

## 7.10 User Story: A-02

### Token Generation with Time Estimates

As a: One stop Admin

I want: to generate unique tokens with estimated timeframes for every student request,

So that: they can be processed on a first come first served basis.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Develop token generation system to create unique tokens for each student request.
- Integrate estimated timeframes calculation for each request based on workload.
- Implement first come first served basis logic for assigning tokens to requests.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.11 User Story: A-03

### Degree Issuance Upon Approval

As a: One stop Admin

I want: to issue degrees to the students upon approvals from all the relevant authorities,

So that: the students can be informed of the completion of their requests and receive their degrees.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Develop functionality to verify relevant approvals before issuing degrees.
- Implement system for notifying students upon degree issuance.
- Ensure accurate record-keeping of degree issuance status.

#### Associated User Stories:

1. User Story A-04: Request Management
2. User Story Y-03: Request Evaluation and Decision – FYP
3. User Story Y-03: Request Evaluation and Decision – Finance
4. User Story F-01: Verification of Outstanding Fees

## 7.12 User Story: A-04

### Request Management

As a: One stop Admin

I want: to view the status of all pending, processed, and new requests,

So that: I can manage and prioritize them effectively.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Develop interface to display pending, processed, and new requests.
- Implement filtering and sorting functionality for requests based on status and submission date.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.13 User Story: A-05

### Transcript Generation

As a: One stop Admin

I want: the transcripts to automatically be generated for students when their degree issuance request is approved,

So that: the students can receive their academic documentation.

### Context:

This story is related to the sprint-03.

### Tasks:

- Implement automated transcript generation process upon approval of degree issuance request.
- Integrate transcript generation with degree issuance system.

### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.14 User Story: A-06

### Handling Request Objections

As a: One stop Admin

I want: to know about any objections raised by the finance or FYP departments regarding a student's degree issuance,

So that: I can inform the student of the issue and guide them about the next steps.

### Context:

This story is related to the sprint-02.

### Tasks:

- Implement system for tracking objections raised by finance or FYP departments regarding degree issuance.
- Create a notification mechanism for admins to be alerted about objections.
- Develop process for informing students about objections and guiding them through resolution steps.

### Associated User Stories:

1. User Story F-02: Requests Evaluation and Decision - Finance
2. User Story Y-03: Request Evaluation and Decision – FYP

## 7.15 User Story: A-07

### Automatic List Updates

As a: One stop Admin

I want: the requests list to automatically be updated when a degree is issued,

So that: it is clear which requests have been processed.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Implement automatic update mechanism for requests list upon degree issuance.
- Ensure synchronization between degree issuance process and requests list.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission
2. User Story A-03: Degree Issuance Upon Approval
3. User Story F-02: Requests Evaluation and Decision - Finance
4. User Story Y-03: Request Evaluation and Decision – FYP

## 7.16 User Story: D-01

### Filter Requests by Day

As a: Director

I want: to view a list of all the requests generated on a specific day,

So that: I have a comprehensive overview of daily activities.

#### Context:

This story is related to the sprint-03.

#### Tasks:

- Develop a filter mechanism to sort requests based on the specified day.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission
2. User Story A-04: Request Management



## **7.17 User Story: D-02**

### **View Pending Requests**

As a: Director

I want: to access a list of all pending requests,

So that: I can monitor the remaining workload and identify bottlenecks in processing.

#### **Context:**

This story is related to the sprint-01.

#### **Tasks:**

- Develop a filter mechanism to filter requests based on status.

#### **Associated User Stories:**

1. User Story A-04: Request Management

## **7.18 User Story: D-03**

### **View Processed Requests**

As a: Director

I want: to view all the processed requests,

So that: I can track the efficiency of the process.

#### **Context:**

This story is related to the sprint-03.

#### **Tasks:**

- Integrate filter by status "processed" to the requests filter mechanism.

#### **Associated User Stories:**

1. User Story A-04: Request Management
2. User Story A-03: Degree Issuance Upon Approval

## 7.19 User Story: D-04

### View Processing Metrics

As a: Director

I want: to view processing details such as the time taken by each department to process a request,

So that: I can identify bottlenecks and optimize the processes accordingly.

#### Context:

This story is related to the sprint-03.

#### Tasks:

- Implement a system to capture and store department-wise processing time data for each request.
- Develop a dashboard or reporting mechanism for the Director to view processing metrics.
- Design visualization interfaces for the analysis and identification of bottlenecks.

#### Associated User Stories:

1. User Story A-04: Request Management
2. User Story Y-01: Request Processing Timestamps

## 7.20 User Story: D-05

### Active Progress Tracking - Director

As a: Director

I want: to track the real-time progress of requests,

So that: I can stay informed about the progress and status of every request.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Design a real-time progress tracking system for all requests.
- Implement a dashboard for the Director to monitor the progress of requests.
- Ensure synchronization and accuracy of real-time data updates across the system.

#### Associated User Stories:

1. User Story S-03: Query Progress Tracking

## 7.21 User Story: Y-01

### Request Processing Timestamps

As a: FYP Department Staff

I want: each request to be time stamped when the processing begins and ends,

So that: accurate record of processing durations is maintained.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Implement a system to timestamp each request when processing begins and ends
- Develop functionality to calculate and display processing durations based on timestamps.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.22 User Story: Y-02:

### Degree Issuance Request Notification - FYP

As a: FYP Department Staff

I want: to receive notifications when a student submits a degree issuance request,

So that: I can promptly begin processing their queries.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Integrate degree issuance requests for FYP staff into the notification system.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.23 User Story: Y-03

### Request Evaluation and Decision - FYP

As a: FYP Department Staff

I want: to accept, reject, or raise objections to a degree issuance request,

So that: I can effectively process requests and inform the student of the decision.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Design a user interface for FYP Department Staff to review and evaluate degree issuance requests.
- Implement functionality for staff members to accept, reject, or raise objections to each request.
- Develop a system for storing comments or reasons provided by staff members for their decisions.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.24 User Story: F-01

### Verification of Outstanding Fees

As a: Finance Department Staff

I want: to verify that all outstanding fees of a student including the issuance fees are paid,

So that: I can ensure that an issuance request meets the financial criteria for approval.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Develop a UI for Finance Department Staff to verify fee payment status of students.
- Ensure that staff members can easily identify whether the financial criteria for approval are met.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission
2. User Story S-07: Payment of Degree Issuance Request Fees

## 7.25 User Story: F-02

### Requests Evaluation and Decision - Finance

As a: Finance Department Staff

I want: to accept, reject, or raise objections to a degree issuance request,

So that: I can effectively process requests and inform the student of the decision.

#### Context:

This story is related to the sprint-01.

#### Tasks:

- Design a user interface for Finance Department Staff to review and evaluate degree issuance requests.
- Implement functionality for staff members to accept, reject, or raise objections to each request.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 7.26 User Story: F-03

### Degree Issuance Request Notification - Finance

As a: Finance Department Staff

I want: to receive notifications when a student submits a degree issuance request,

So that: I can promptly begin processing their queries.

#### Context:

This story is related to the sprint-02.

#### Tasks:

- Integrate degree issuance requests for Finance staff into the notification system.

#### Associated User Stories:

1. User Story S-01: Degree Issuance Request Submission

## 8. Software Testing

Note: No. of testcases for weak and strong equivalence classes are represented by **w** and **s**

### 8.1 Login Page

#### Equivalence Class Partitions:

Input Field	Equivalence Class	Validity	w	s
Email	Contains exactly one "@" symbol Format: [local part]@[domain part]	Valid	4	10
	Missing "@" symbol Multiple "@" symbols Incorrect format (e.g., no domain part)	Invalid		
Password	8 or more characters	Valid	2	3
	Less than 8 characters	Invalid		

#### Boundary Value Analysis and Testcases:

##### 1. Email:

##### Boundary Value Analysis

- Lower Bound: "a@a.a"

This is the smallest valid email address, with a local part of one character, a domain part of one character, and a top-level domain of one character.

- Upper Bound: "a" repeated 30 times + "@example.com"

This is an email address with 65 characters in the local part, which is the maximum length allowed by most email systems, followed by a standard domain part.

- Near Lower Bound: "a@example.com"

This is a valid email address with a local part of one character and a standard domain part.

- Near Upper Bound: "a" repeated 29 times + "@example.com"

This is an email address with 64 characters in the local part, which is just below the maximum length allowed by most email systems, followed by a standard domain.

##### Test Cases:

- Valid: "user@example.com", "first.last@example.co.uk"
- Invalid: "user", "user@", "user@@example.com"

## 2. Password:

### Boundary Value Analysis

- Lower Bound: "12345678"  
This is the smallest valid password, with exactly 8 characters.
- Upper Bound: "a" repeated 64 times.  
This is a password with 64 characters, which is a maximum valid length.
- Near Lower Bound: "1234567a"  
This is a password with 7 characters followed by one additional character.
- Near Upper Bound: "a" repeated 63 times + "1"  
This is a password with 63 characters followed by one additional character.

### Test Cases:

- Valid: "password1", "secure123", "longpassword1234"
- Invalid: "pass", "1234567", "a" repeated 65 times

## 8.2 Issuance Form

### Equivalence Class Partitions:

Input Field	Equivalence Class	Validity	w	s
Phone Number	Exactly 10 digits (Range is 11-15 digits)	Valid	4	10
	Less than 10 digits More than 10 digits, Non-numeric characters	Invalid		
Email	Contains exactly one "@" symbol Format: [local part]@[domain part]	Valid	4	10
	Missing "@" symbol Multiple "@" symbols Incorrect format (e.g., no domain part)	Invalid		
Document Type	"Transcript", "Degree"	Valid	2	3
	"nothing chosen", NULL	Invalid		
Start and End Date	Dates between 1900-01-01 and 2028-12-31 Format YYYY-MM-DD.	Valid	4	10
	Dates outside the range of 1900-01-01 to 2028-12-31. Dates in incorrect formats (e.g., not YYYY-MM-DD).	Invalid		

## Boundary Value Analysis and Testcases:

### 1. Phone Number

#### Boundary Value Analysis

- Lower Bound: 9 digits
- Upper Bound: 16 digits
- Near Lower Bound: 10 digits
- Near Upper Bound: 15 digits

#### Test Cases:

- Valid: "1234567890", "123456789012345", "1234567890123456"
- Invalid: "123", "12345678901234567", "123abc456"

### 2. Document Type

#### Boundary Value Analysis

NA

#### Test Cases:

- Valid Input: "Transcript", "Degree"
- Invalid Input: "", NULL

### 3. Start and End Date

#### Boundary Value Analysis

- Lower Bound: 1899-12-31. One day before the valid date range.
- Upper Bound: 2028-12-31. The last date in the valid range.
- Near Lower Bound: 1900-01-01. The first date in the valid range.
- Near Upper Bound: 2028-12-30. One day before the upper bound.

#### Test Cases:

- Valid Dates: "1900-01-01", "2000-02-29", "2028-12-31"
- Invalid Dates: "1899-12-31", "2028-12-32", "2000-02-30", "2020/01/01"



### 8.3 Complaint Form

#### Equivalence Class Partitions:

Input Field	Equivalence Class	Validity	w	s
Description	Text length less than or equal to 45 characters	Valid	2	3
	Text length greater than 45 characters	Invalid		
Document Type	"Transcript", "Degree"	Valid	2	3
	"nothing chosen", NULL	Invalid		

#### Boundary Value Analysis and Testcases:

##### 1. Description

###### Boundary Value Analysis

- Lower Bound: "Short description"  
This is the smallest valid text, with 17 characters.
- Upper Bound: "A" repeated 45 times  
This is a text with 45 characters, which is the maximum allowed length.
- Near Lower Bound: "Very short description"  
This is a valid text with 23 characters, just above the lower bound.
- Near Upper Bound: "A" repeated 44 times + "a"  
This is a text with 45 characters, just below the upper bound.

###### Test Cases:

- Valid: "Brief overview", "Lorem ipsum dolor sit amet", "Description"
- Invalid: "This is a very long description that exceeds the maximum limit of 45 characters"

##### 4. Document Type

###### Boundary Value Analysis

NA

###### Test Cases:

- Valid Input: "Transcript", "Degree"
- Invalid Input: "", NULL

## 8.4 Payments

### Equivalence Class Partitions:

Input Field	Equivalence Class	Validity	w	s
Credit Card Number	16-digit number	Valid	4	10
	Less than 16 digits More than 16 digits Contains Alphabets or Special Characters	Invalid		
CCV	3-digit number	Valid	4	10
	Less than 3 digits More than 3 digits Contains Alphabets or Special Character	Invalid		
Expiry Date	Dates between 1900-01-01 and 2028-12-31 Format YYYY-MM-DD.	Valid	4	10
	Dates outside the range of 1900-01-01 to 2028-12-31. Dates in incorrect formats (e.g., not YYYY-MM-DD).	Invalid		
Phone Number	10-digit number in range(11-15 digits)	Valid	4	10
	Less than 10 digits More than 15 digits Non-numeric characters	Invalid		

### Boundary Value Analysis and Testcases:

#### 1. Credit Card Number

##### Boundary Value Analysis

- Lower Bound: "0000000000000000"  
This is the smallest valid credit card number, with all digits being 0.
- Upper Bound: "9999999999999999"  
This is the largest possible 16-digit number.
- Near Lower Bound: "0000000000000001"  
This is a valid credit card number with the smallest possible non-zero value.
- Near Upper Bound: "9999999999999990"  
This is a credit card number with the largest possible value within the 16-digit limit.

##### Test Cases:

- Valid: "4111111111111111", "1234567890123456"
- Invalid: "1234", "41111111111111112", "abcdeabcdeabcdea"

## 2. CCV

### Boundary Value Analysis

- Lower Bound: "000"  
This is the smallest valid CCV, with all digits being 0.
- Upper Bound: "999"  
This is the largest valid CCV, with all digits being 9.
- Near Lower Bound: "001"  
This is a valid CCV with the smallest possible non-zero value.
- Near Upper Bound: "998"  
This is a valid CCV with the largest possible value that is less than 999.

### Test Cases:

- Valid: "123", "456", "999"
- Invalid: "12", "1234", "abc"

## 3. Phone Number

Same as in user story S-01

## 4. Expiry Date

Same as Start and End date in user story S-01

## 8.5 Objections to Requests

### Equivalence Class Partitions:

Input Field	Equivalence Class	Validity	w	s
Description	Text length less than or equal to 45 characters	Valid	2	3
	Text length greater than 45 characters	Invalid		

### Boundary Value Analysis and Testcases:

#### 1. Description

Same as Description in user story S-02

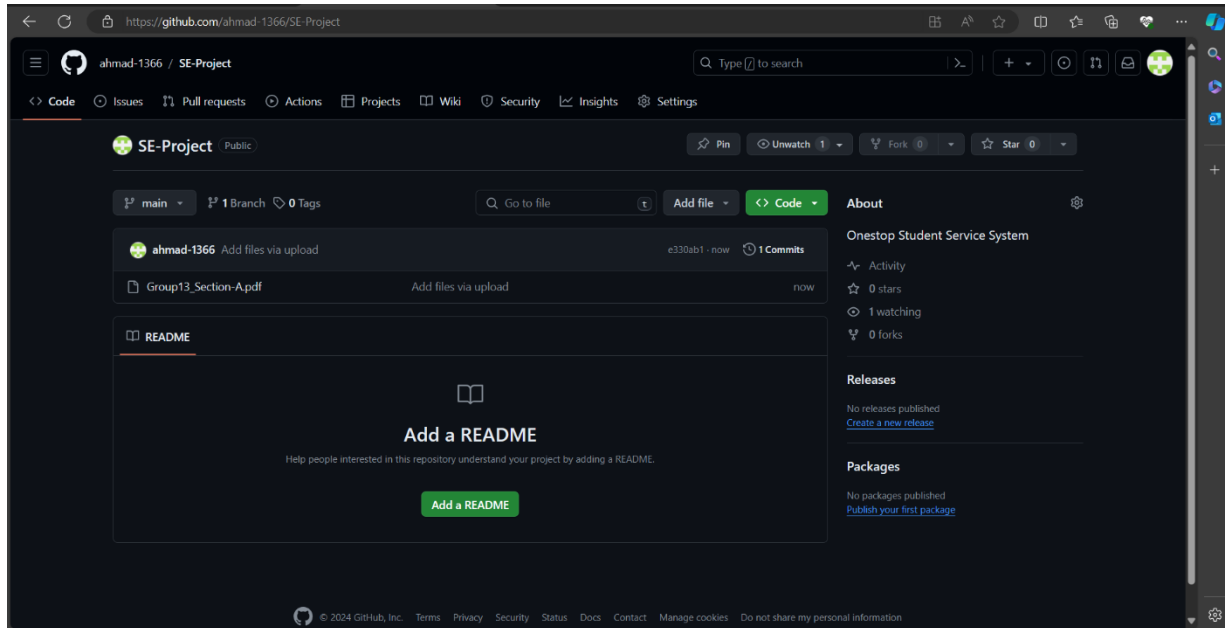
## Appendix A: Glossary

<b><i>Degree Issuance System</i></b>	A web-based application for issuing degrees to students.
<b><i>One Stop Services Centre</i></b>	A centralized facility providing student assistance.
<b><i>Degree Issuance Form</i></b>	Digital form for submitting degree requests.
<b><i>Complaint Form</i></b>	Digital form for reporting document errors.
<b><i>Activity Tracking</i></b>	Monitoring request status.
<b><i>Token Generation</i></b>	Creating unique request identifiers.
<b><i>Request Management</i></b>	Interface for admin oversight of requests.
<b><i>Degree and Transcript Issuance</i></b>	Process of providing degree documents.
<b><i>DBMS</i></b>	Software for storing and managing data.
<b><i>Web Server</i></b>	Application hosting the system interface.
<b><i>Notification Services</i></b>	External tools for user alerts.
<b><i>Payment Gateway</i></b>	System for verifying fee payments.
<b><i>HTTP/HTTPS</i></b>	Protocols for web data transmission.
<b><i>SMTP/SMTPS</i></b>	Protocols for email transmission.
<b><i>SMS Gateway</i></b>	Service for sending text message notifications.

## Appendix B: Analysis Models

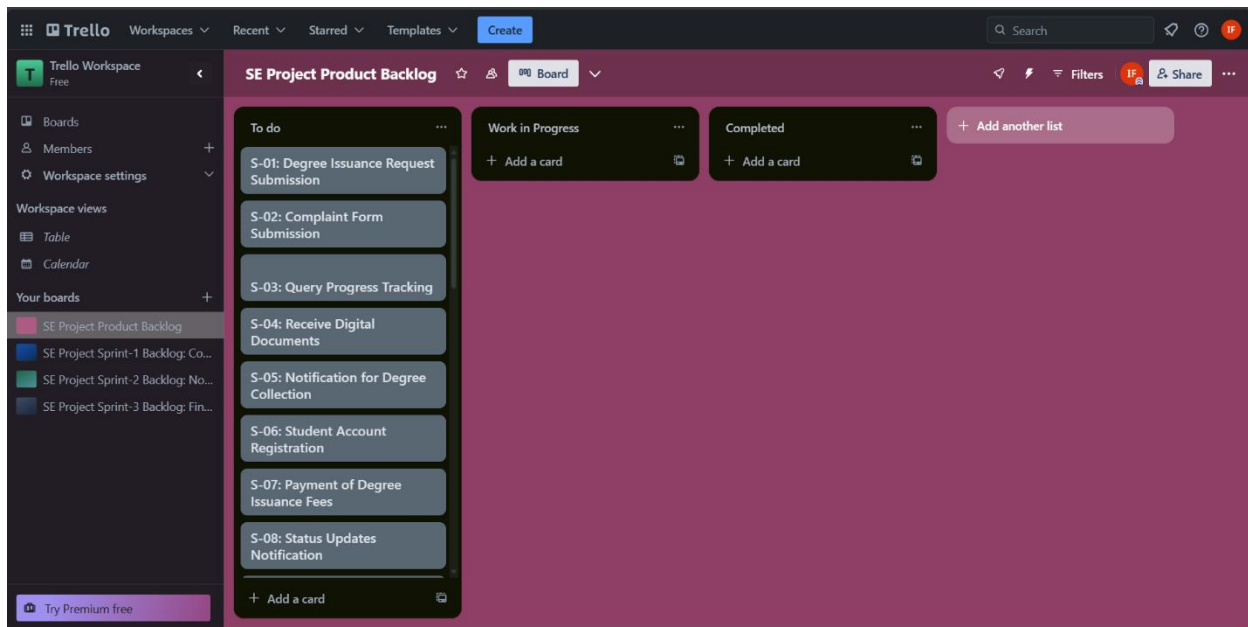
### GitHub

[ahmad-1366/SE-Project \(github.com\)](https://github.com/ahmad-1366/SE-Project)

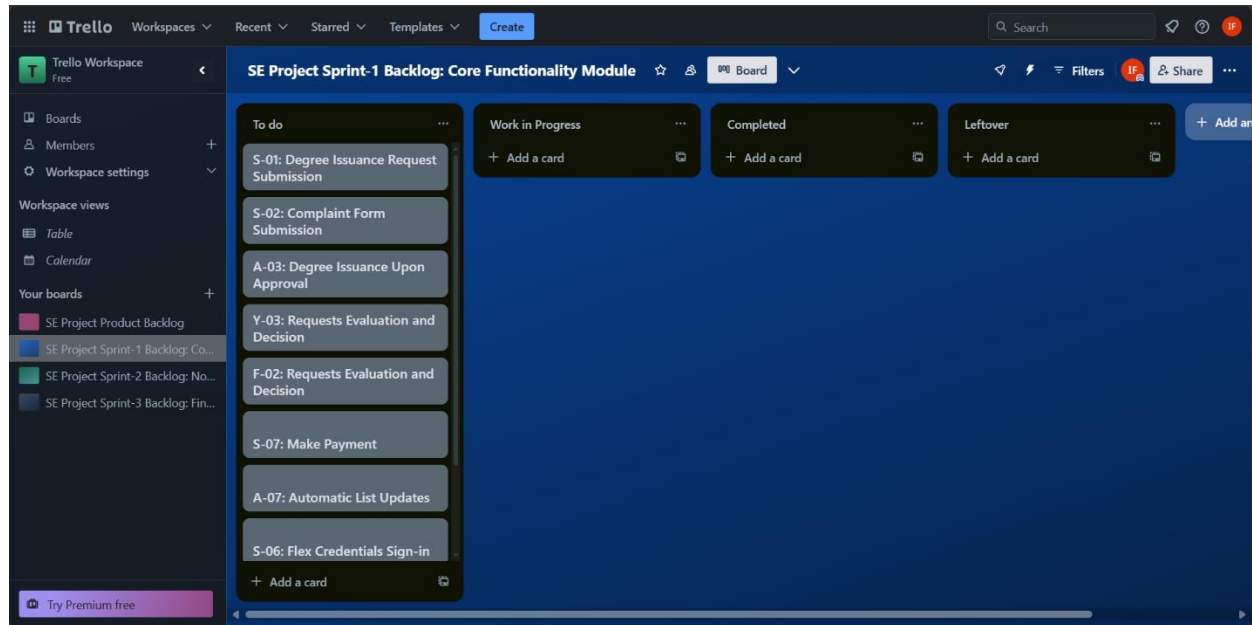


### Trello

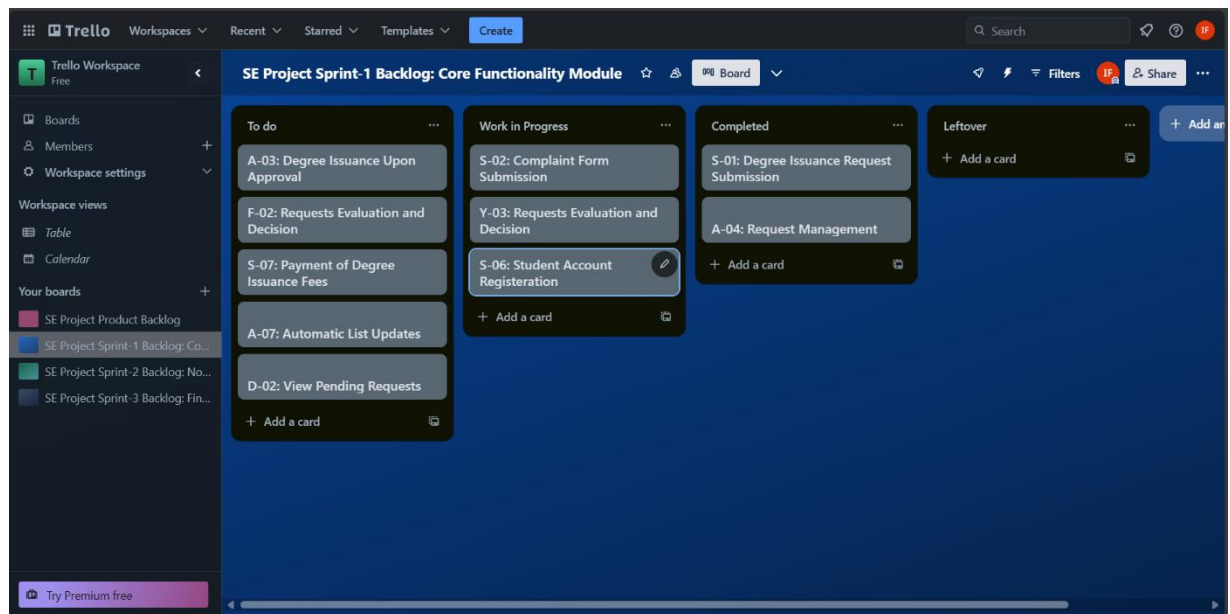
#### Product Backlog



## Sprint Backlog

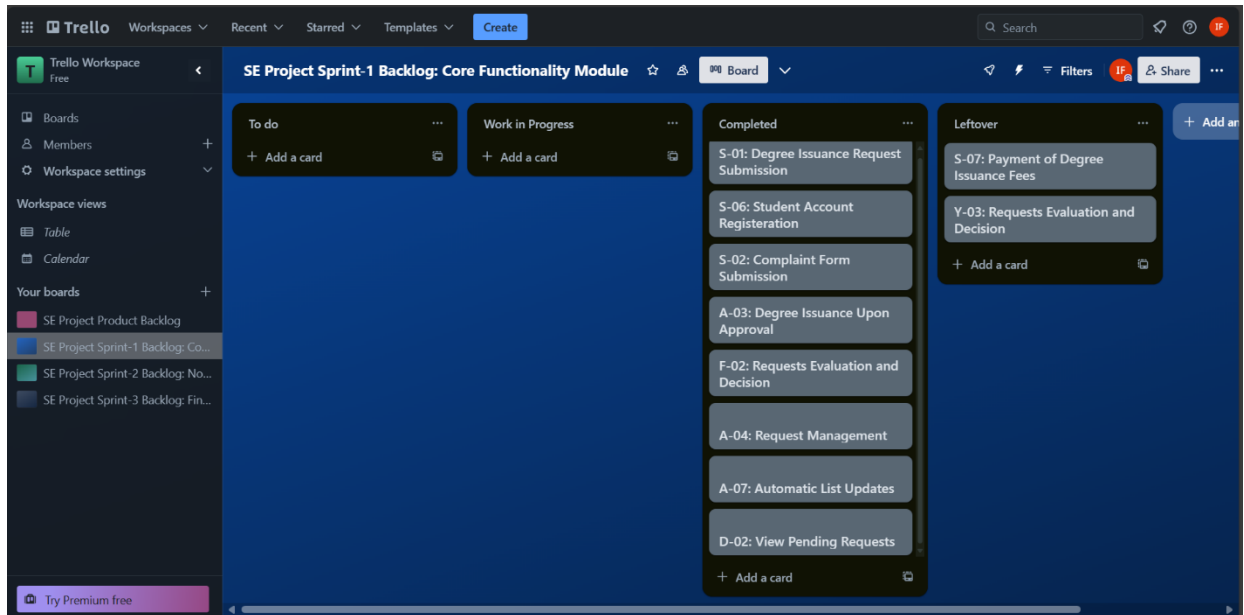


## Sprint In Progress

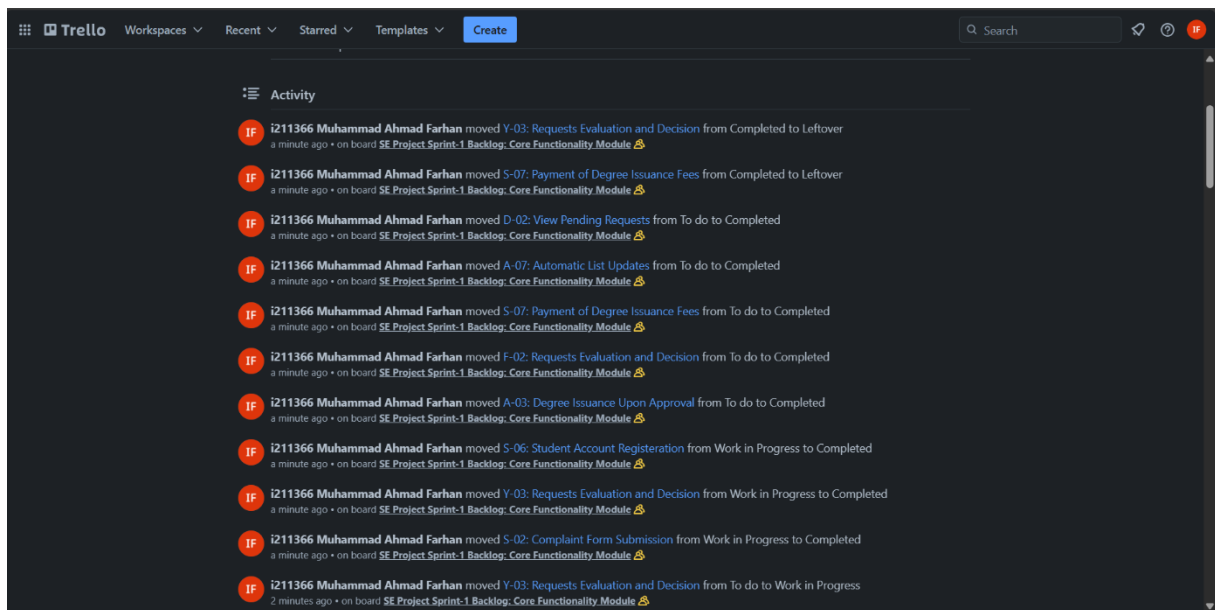


## Software Requirement Specification for Onestop System

### Sprint Completed

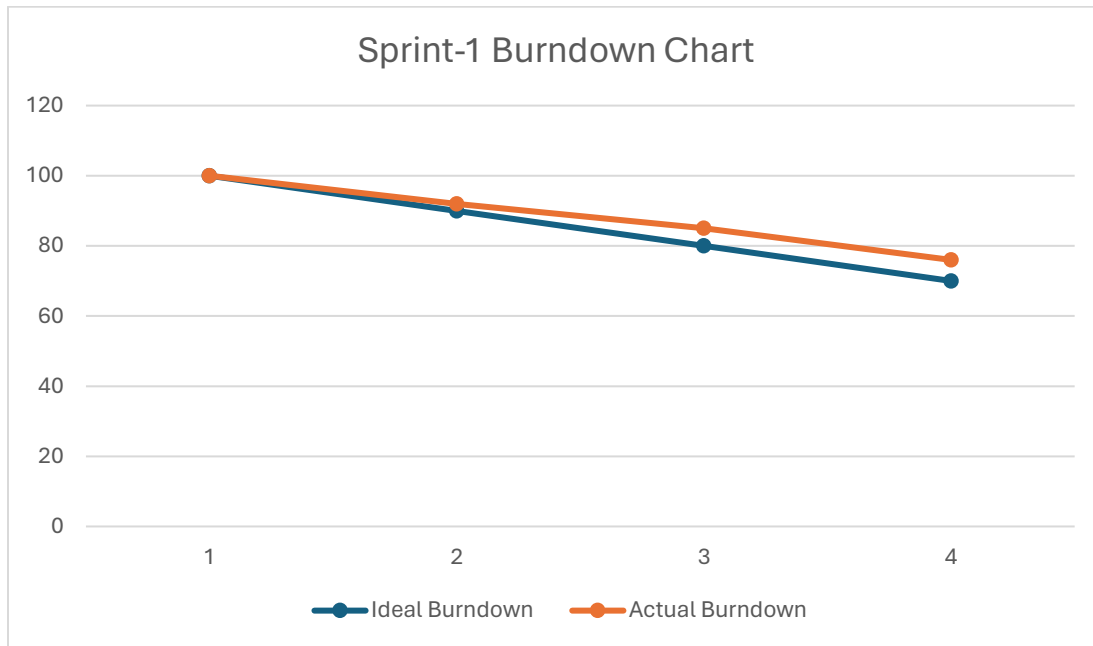


### Activities Log

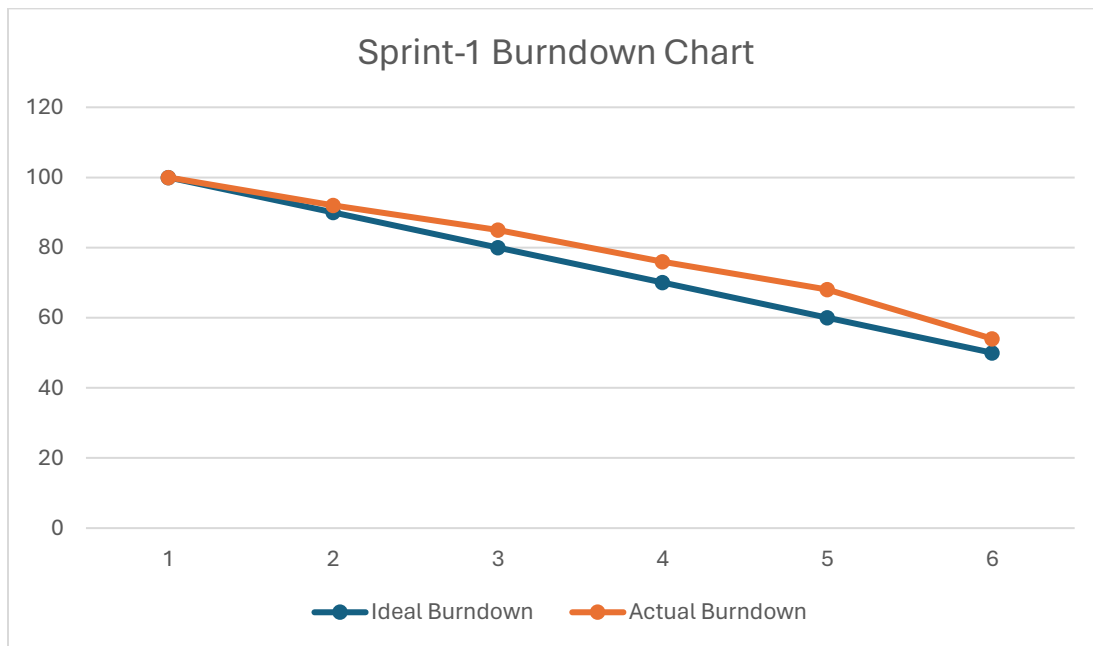


## Sprint Burndown Chart

### Snapshot-1

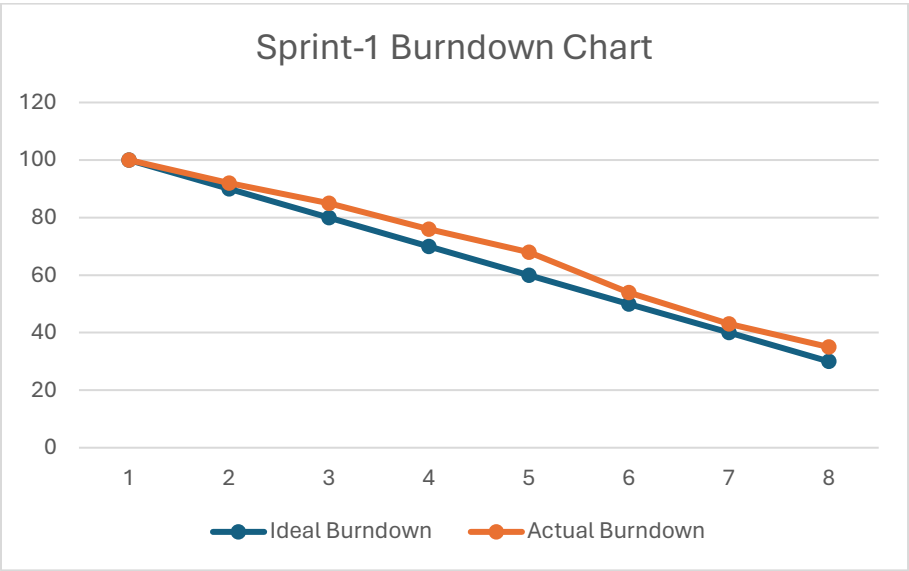


### Snapshot-2

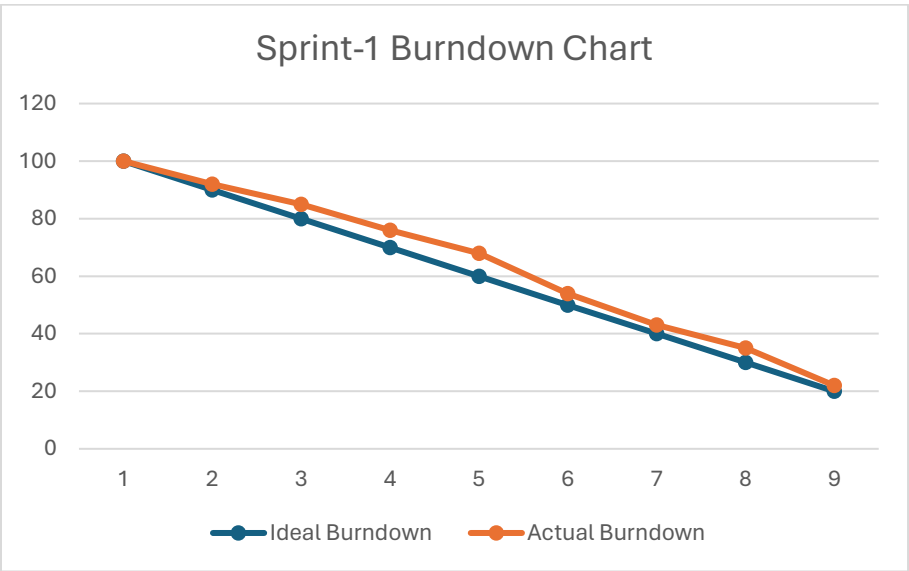




Snapshot-3



Snapshot-4

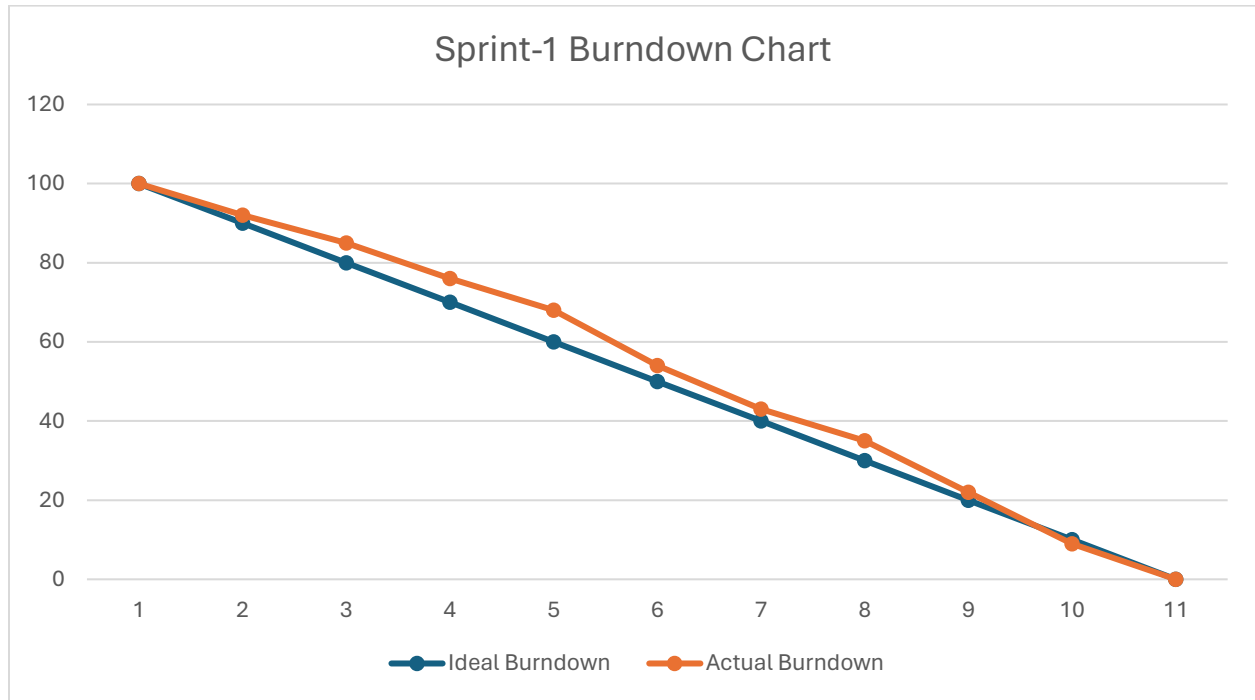


Sprint Timeline

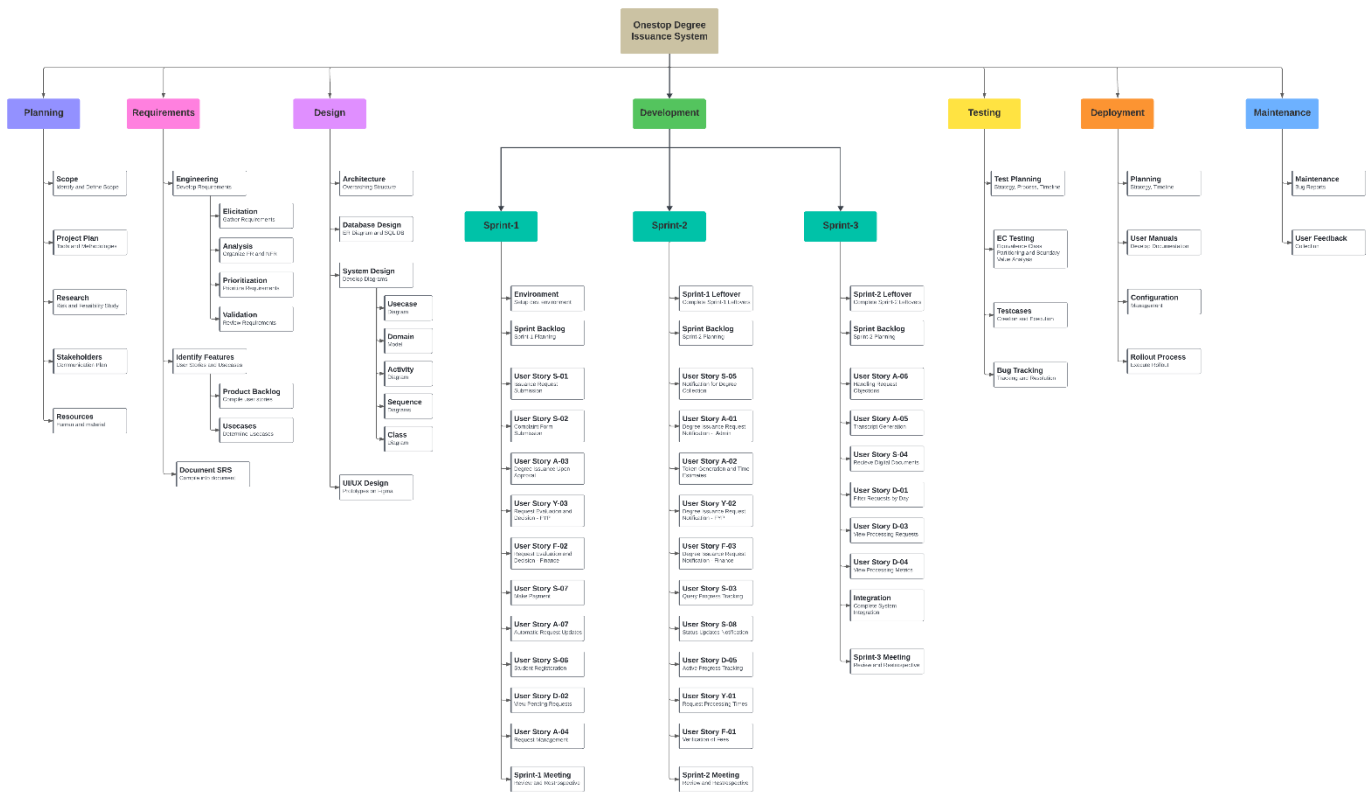
Sprint -1 Burndown Chart											
User Story ID	User Story Name	Sprint Timeline									
		15th April, 2024	16th April, 2024	17th April, 2024	18th April, 2024	19th April, 2024	22th April, 2024	23th April, 2024	24th April, 2024	25th April, 2024	27th April, 2024
User Story S-01	Degree Issuance Request Submission	15	11	9	5	0	0	0	0	0	0
User Story S-02	Complaint Form Submission	8	8	8	8	8	2	0	0	0	0
User Story A-03	Degree Issuance Upon Approval	8	8	8	8	8	6	3	0	0	0
User Story Y-03	Request Evaluation and Decision - FYP	10	10	10	10	10	10	10	10	6	2
User Story F-02	Request Evaluation and Decision - Financ	14	14	14	14	14	14	14	14	10	4
User Story S-07	Payment of Degree Issuance Fees	18	18	18	18	18	12	8	5	0	0
User Story A-07	Automatic List Updates	6	6	6	6	6	6	6	6	6	3
User Story S-06	Student Account Registrations	5	5	2	0	0	0	0	0	0	0
User Story D-02	Request Management	12	8	6	3	0	0	0	0	0	0
User Story A-04	View Pending Requests	4	4	4	4	4	4	2	0	0	0
Ideal Estimated Remaining Working Hours		100	90	80	70	60	50	40	30	20	10
Likely Estimated Remaining Work Hours		100	92	85	76	68	54	43	35	22	9

## Software Requirement Specification for Onestop System

## Snapshot-5

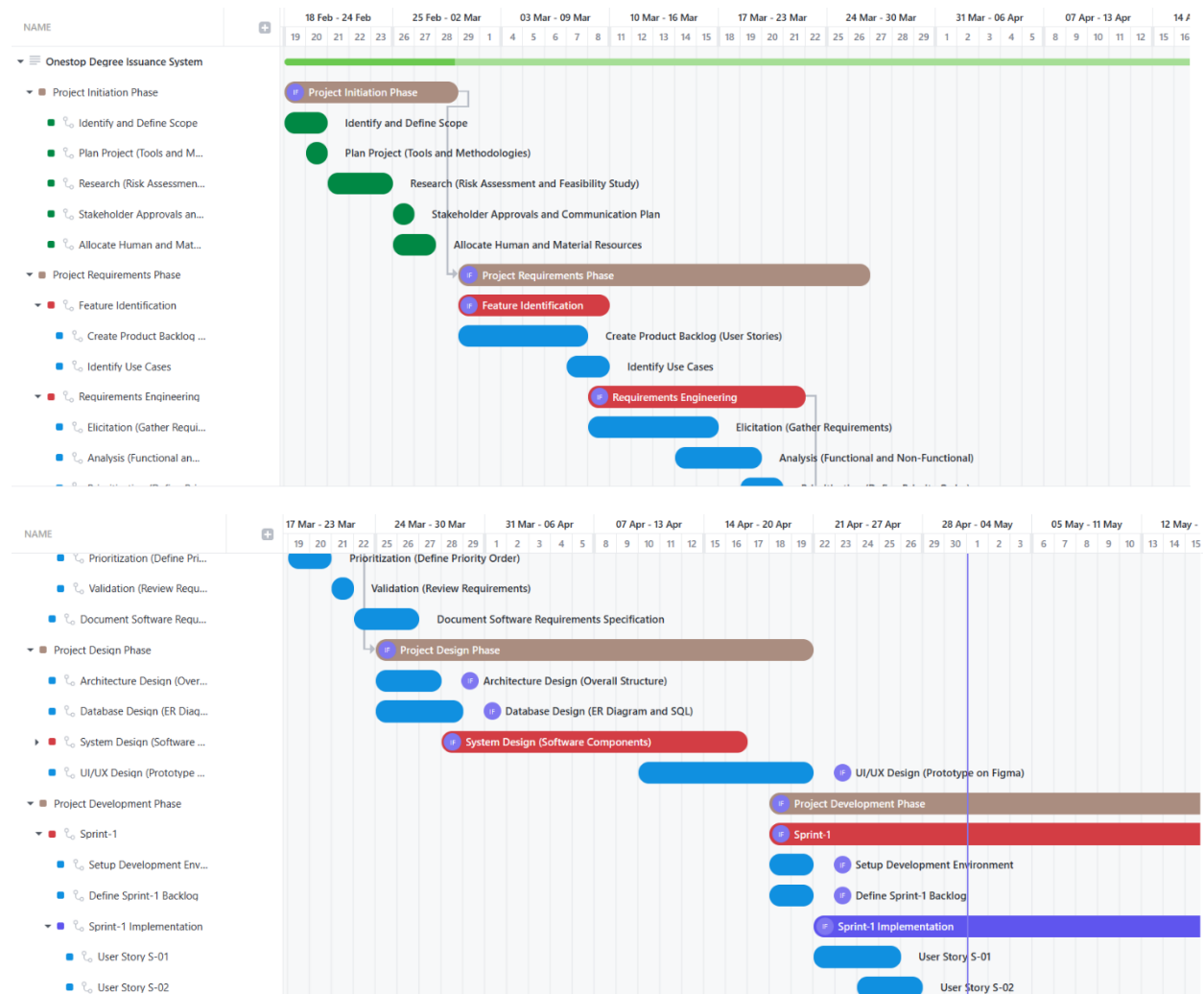


## Work Breakdown Structure



## Gantt Chart

<https://sharing.clickup.com/9018255734/g/h/8crfabp-38/d9f926c845c34a6>



## Appendix C: To Be Determined List

None as of yet.