

---

# **Software Requirements Specification**

**for**

## **Onestop Degree Issuance Module**

**Version 1.0 approved**

**Prepared by**

**21I-2992 Daniyal Kaleem Sheikh**

**21I-0822 Abdullah Masood**

**21I-2478 Muhammad Tayyab Sohail**

**FAST NUCES**

**April 2nd, 2024**

# Table of Contents

<b>1. Introduction.....</b>	<b>4</b>
1.1 Purpose.....	4
1.2 Document Conventions.....	4
1.3 Intended Audience and Reading Suggestions.....	4
1.4 Product Scope.....	6
1.5 References.....	7
<b>2. Overall Description.....</b>	<b>7</b>
2.1 Product Perspective.....	7
2.2 Product Functions.....	7
2.3 User Classes and Characteristic.....	7
2.4 Operating Environment.....	9
2.5 Design and Implementation Constraints.....	10
2.6 User Documentation.....	10
2.7 Assumptions and Dependencies.....	11
<b>3. External Interface Requirements.....</b>	<b>11</b>
3.1 User Interfaces.....	11
3.2 Hardware Interfaces.....	12
3.3 Software Interfaces.....	13
3.4 Communications Interfaces.....	14
<b>4. System Features.....</b>	<b>14</b>
4.1 Degree Issuance Form.....	14
4.2 Complaint Form.....	15
4.3 Feedback Form.....	16
4.4 Activity Tracking for Students.....	16
4.5 Degree Receipt for Students.....	17
4.6 FYP Department - Student Notification and Decision Making.....	18
4.7 Finance Department - Student Notification and Payment Verification.....	19
4.8 Decision Making for Finance Department.....	20
4.9 Student Notification Management and Token Generation for One Stop Admin.....	20
4.10 Request Management Dashboard for One Stop Admin.....	21
4.11 Degree Issuance and Transcript Generation for One Stop Admin.....	22
4.12 Request Monitoring and Activity Tracking Dashboard for Director.....	23
<b>5. Other Nonfunctional Requirements.....</b>	<b>24</b>
5.1 Performance Requirements.....	24
5.2 Safety Requirements.....	24

5.3 Security Requirements.....	25
5.4 Software Quality Requirements.....	26
5.5 Business Rules.....	28
6. Diagrams.....	29
6.1 Use Case Diagram.....	29
6.2 Activity Diagram.....	29
6.3 Sequence Diagram.....	31
<b>7. Trello Board.....</b>	<b>33</b>
<b>8. Github.....</b>	<b>35</b>
<b>Appendix A: Glossary.....</b>	<b>35</b>
<b>Appendix B: Analysis Models.....</b>	<b>35</b>
<b>Appendix C: To Be Determined List.....</b>	<b>35</b>

## Revision History

Name	Date	Reason For Changes	Version

# **1. Introduction**

## **1.1 Purpose**

This Software Requirements Specification (SRS) document specifies the requirements for the development of the One Stop Services Centre System for the FAST Islamabad campus. This document covers the initial release of the system and outlines the scope of the product, which includes modules for degree issuance, complaint handling, activity tracking, payment management, and notification systems. Additionally, it defines the roles and responsibilities of various users such as students, FYP department, Finance department, One Stop Admin, and Director within the system. This SRS provides a comprehensive overview of the features and functionality required for the successful implementation of the One Stop Services Centre System.

## **1.2 Document Conventions**

This Software Requirements Specification (SRS) follows the following conventions:

1. **Priority Levels:** Each requirement statement is accompanied by a priority level indicating its importance and urgency. The priority levels used are "High," "Medium," and "Low." These priority levels help stakeholders understand the relative importance of each requirement in relation to others.
2. **Requirement Numbering:** Each requirement statement is uniquely identified with a requirement number or tag. This numbering system allows for easy reference and cross-referencing of requirements within the document.
3. **Formatting for Clarity:** The document is formatted to enhance readability, with clear spacing, indentation, and alignment. Bulleted or numbered lists are used where applicable to present information in a concise and structured manner.
4. **Consistent Terminology:** Consistent terminology and language are used throughout the document to ensure clarity and avoid ambiguity. Technical terms and acronyms are defined in the glossary section for reference.
5. **Visual Aids:** Where necessary, visual aids such as diagrams or tables may be included to illustrate concepts, system components, or data flows. These visual aids complement the textual descriptions and enhance understanding.
6. **References and Citations:** Any references to external documents, standards, or guidelines are cited appropriately within the text. This ensures transparency and facilitates further exploration of related topics by stakeholders.

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

This Software Requirements Specification (SRS) document is intended for a diverse audience involved in the development, management, testing, and use of the One Stop Services Centre System. The document is relevant for:

1. **Developers:** Developers will utilize this document to understand the functional and non-functional requirements of the system, guiding them in the implementation phase.
2. **Project Managers:** Project managers will use the SRS to oversee the development process, ensuring that the project aligns with the specified requirements and objectives.
3. **Users:** Users, primarily students and administrative staff at FAST Islamabad campus, will refer to the SRS to understand the features and functionality of the system, as well as their roles and responsibilities within it.
4. **Testers:** Testers will utilize the SRS to develop test cases and scenarios, ensuring comprehensive testing of the system to validate its functionality and reliability.
5. **Documentation Writers:** Documentation writers will use the SRS as a reference to create user manuals, technical documentation, and training materials for end-users and system administrators.

The rest of this SRS contains detailed information organized into the following sections:

1. **Overview Sections:** Begin by reading the introduction and scope sections to understand the purpose and scope of the software project.
2. **Functional Requirements:** Review the detailed functional requirements section to understand the specific features and functionalities of the degree issuance module.
3. **User Documentation:** If applicable, refer to the user documentation section for guidance on using the software and understanding its features.
4. **External Interface Requirements:** Understand the interfaces between the software and other systems or components, including communication protocols and data formats.
5. **Nonfunctional Requirements:** Pay attention to performance, safety, security, and other nonfunctional requirements to ensure the software meets quality standards.
6. **User Classes and Characteristics:** Identify the different user classes and their characteristics to understand their needs and requirements.
7. **Assumptions and Dependencies:** Be aware of any assumptions or dependencies that could impact the project's success or timeline.
8. **Operating Environment:** Understand the hardware and software environment in which the software will operate to ensure compatibility and smooth deployment.
9. **Overall Description:** Gain insight into the context and origin of the product, including its relationship to other system components.
10. **Glossary:** Refer to the glossary for definitions of terms and acronyms used throughout the document.

To ensure a comprehensive understanding of the document, readers are recommended to follow this sequence. This suggested sequence will facilitate a thorough understanding of the One Stop Services Centre System and its associated requirements for each type of reader.

## 1.4 Product Scope

The software being specified is the "One Stop Services - Degree Issuance Module," designed to streamline and automate the process of issuing degrees to students at FAST Islamabad campus. Its purpose is to centralize and digitize degree issuance procedures, providing students with a convenient platform to request, track, and receive their degrees, while also enabling administrators to efficiently manage and process these requests.

### **Key Benefits, Objectives, and Goals:**

**Efficiency:** By digitizing and automating degree issuance processes, the software aims to reduce manual effort and paperwork, leading to faster turnaround times for degree issuance.

**Transparency:** Students can track the status of their degree issuance requests in real-time, enhancing transparency and reducing uncertainty in the process.

**Accuracy:** The system ensures accuracy in data capture and fee calculations, minimizing errors and discrepancies in degree issuance.

**Improved Student Experience:** Providing a user-friendly online platform for degree issuance enhances the overall student experience, making it easier and more convenient for students to obtain their degrees.

**Administrative Effectiveness:** Admins benefit from streamlined workflows, automated notifications, and comprehensive audit logs, enabling them to manage degree issuance requests more effectively.

### **Relation to Corporate Goals or Business Strategies:**

**Enhancing Student Services:** The software aligns with the institution's goal of improving student services by providing a centralized platform for degree issuance, addressing a critical aspect of the student experience.

**Operational Efficiency:** By digitizing degree issuance processes, the software contributes to operational efficiency and cost-effectiveness, supporting the institution's broader strategy of optimizing administrative workflows.

**Reputation and Competitiveness:** A streamlined and efficient degree issuance process reflects positively on the institution's reputation and competitiveness, positioning it as a leader in providing excellent student services and administrative support.

## 1.5 References

1. IEEE 830 Standard for SRS  
<http://www.cse.uaa.alaska.edu/~afkjm/csce401/IEEE830.pdf>

## **2. Overall Description**

### **2.1 Product Perspective**

The product specified in this Software Requirements Specification (SRS) is a new, self-contained system designed to facilitate the degree issuance process within the educational institution. It is not a replacement for any existing systems but rather a standalone solution aimed at streamlining and automating the degree issuance workflow.

The degree issuance module serves as a vital component within the larger ecosystem of administrative and academic services provided by the educational institution. While it operates independently, it interfaces with other existing systems and departments to fulfill its objectives. These interfaces include integration with student information systems (SIS) for retrieving academic records, financial systems for verifying fee payments, and email servers for sending notifications to stakeholders.

### **2.2 Product Functions**

1. Requesting degree issuance
2. Activity tracking
3. Feedback
4. Automated Notification System
5. Fee Handling
6. Request Management
7. Issue Handling
8. Request Processing
9. Transcript issuance
10. Analytics

### **2.3 User Classes and Characteristic**

1. Student
  - a. Frequency of Use: High, as they will interact with the system to submit degree issuance requests, track request status, and receive degrees.
  - b. Subset of Product Functions Used: Mainly degree issuance form submission, activity tracking, and receipt of degree.
  - c. Technical Expertise: Varied, ranging from basic computer literacy to moderate proficiency.
  - d. Educational Level/Experience: Typically undergraduate or graduate students with varying levels of experience in using online systems.
  - e. Security/Privilege Levels: Limited access to personal data and request statuses; no administrative privileges.
2. FYP Department Staff

- a. Frequency of Use: Moderate, as they will review and provide decisions on degree issuance requests.
  - b. Subset of Product Functions Used: Reviewing student notifications, providing decisions (accept, reject, raise objections), and tracking request processing times.
  - c. Technical Expertise: Moderate to high, as they may need to navigate administrative interfaces and understand degree issuance policies.
  - d. Educational Level/Experience: Typically faculty members or administrative staff with expertise in academic processes.
  - e. Security/Privilege Levels: Higher access privileges compared to students, with the ability to review and approve degree issuance requests.
3. Finance Department Staff
- a. Frequency of Use: Moderate, as they will verify payment statuses and process degree issuance fees.
  - b. Subset of Product Functions Used: Reviewing student notifications, verifying outstanding amounts and fees, providing decisions, and tracking payment statuses.
  - c. Technical Expertise: Moderate, as they need to understand financial processes and systems.
  - d. Educational Level/Experience: Staff with expertise in financial management and accounting practices.
  - e. Security/Privilege Levels: Similar to FYP department staff, with access to financial data and the authority to make decisions regarding fee payments.
4. Onestop Admin
- a. Frequency of Use: High, as they will oversee the entire degree issuance process and interact with all user classes.
  - b. Subset of Product Functions Used: Generating tokens for requests, managing request queues, resolving objections, issuing degrees, and updating request statuses.
  - c. Technical Expertise: High, as they need to manage administrative interfaces, troubleshoot issues, and coordinate with different departments.
  - d. Educational Level/Experience: Administrative staff with experience in managing student services and systems.
  - e. Security/Privilege Levels: Highest access privileges, with authority over system administration and data management.
5. Director
- a. Frequency of Use: Low to moderate, as they will primarily monitor system activities and review overall performance.
  - b. Subset of Product Functions Used: Viewing request statistics, monitoring processing times, and tracking system activities.
  - c. Technical Expertise: Moderate, as they need to interpret system metrics and make strategic decisions based on performance data.
  - d. Educational Level/Experience: Senior management with experience in overseeing academic operations and strategic planning.
  - e. Security/Privilege Levels: Elevated access privileges for monitoring and reporting purposes, but limited involvement in day-to-day operations.



**Important User Classes:** The most important user classes for this product are Students, FYP Department Staff, Finance Department Staff, and One Stop Admin, as they directly contribute to the degree issuance process and rely on the system for their respective tasks. The Director serves as an important stakeholder for monitoring system performance and making strategic decisions but has less frequent interaction with the system compared to other user classes.

## 2.4 Operating Environment

The software will operate in a modern computing environment, consisting of the following components:

1. **Hardware Platform:** The software is designed to run on standard hardware configurations commonly used in educational institutions, including desktop computers, laptops, and servers. The hardware should meet minimum requirements for processing power, memory, and storage capacity to ensure optimal performance.
2. **Operating System:** The software is compatible with multiple operating systems, including:
  - a. Windows: Versions 7, 8, and 10.
3. **Web Browsers:** The software is accessible via web browsers commonly used by users, including:
  - a. Google Chrome: Latest version.
  - b. Mozilla Firefox: Latest version.
  - c. Microsoft Edge: Latest version.
4. **Database Management System (DBMS):** The software relies on a database management system (DBMS) to store and manage data. It is compatible with the following DBMS:
  - a. MySQL: Version 5.7 and above.
  - b. Microsoft SQL Server: Version 2016 and above.
5. **Web Server:** The software is deployed on a web server capable of hosting web applications and supporting server-side technologies such as:
  - a. Apache HTTP Server: Version 2.4 and above.
6. **Programming Languages and Frameworks:** The software is developed using commonly used programming languages and frameworks, including:
  - a. Frontend: HTML5, CSS3, JavaScript (ES6+), React.js
  - b. Backend: C#
7. **Other Software Components:** The software may interact with external systems and services, such as student information systems (SIS), email servers, and payment gateways. Integration with these components should be seamless and well-documented to ensure interoperability and data consistency.
8. **Network Infrastructure:** The software requires a stable network connection to facilitate communication between client devices and server infrastructure. It should be deployed in an environment with sufficient bandwidth and network reliability to support concurrent user access and data transfer.

## 2.5 Design and Implementation Constraints

The software must have English as default language for all users. Other than this constraint on

language, since the software is to be used internally within the university without any regulatory constraints, there is no further requirement of this type left to be fulfilled.

## 2.6 User Documentation

The user documentation components that will be delivered along with the One Stop Services Centre System include:

1. **User Manuals:** Comprehensive manuals will be provided to guide users through the functionalities of the system. These manuals will include step-by-step instructions, screenshots, and explanations.
2. **Tutorials:** Tutorials will be developed to provide users with hands-on training on how to use the system efficiently. These tutorials may include video demonstrations, walkthroughs, and interactive exercises to help users familiarize themselves with the system's interface and workflows.
3. **FAQs (Frequently Asked Questions):** A compilation of frequently asked questions and their answers will be provided to address common queries and concerns that users may encounter while using the system.
4. **Glossary:** A glossary of terms and terminology used within the system will be included. This glossary will define key terms, acronyms, and technical jargon.
5. **Release Notes:** Release notes will accompany each software update to inform users about new features, enhancements, bug fixes, and any other changes implemented in the system. These notes will provide users with insights into the latest developments and improvements in the system.

### Delivery Formats or Standards:

1. **PDF:** User manuals and tutorials may be delivered in PDF format for easy access and printing.
2. **Plain Text:** FAQs and glossary entries may be delivered in plain text format for simplicity and accessibility.
3. **Online Portal:** User documentation may be accessible through an online portal or knowledge base hosted on the institution's website for easy access and updates.

## 2.7 Assumptions and Dependencies

1. **Third-Party Components:** The project assumes the availability and compatibility of third-party components or libraries for specific functionalities, such as user authentication, data encryption, and payment processing. Any changes or disruptions to these components could affect the development timeline and functionality of the system.
2. **Operating Environment:** It is assumed that the system will be deployed in a stable and secure operating environment with adequate hardware resources and network infrastructure. Changes to the operating environment, such as updates to underlying software platforms or changes in network configurations, could impact system performance and reliability.
3. **Integration with External Systems:** The project depends on successful integration with external systems or services, such as student databases, financial systems, and email servers,

for data exchange and communication. Any discrepancies or inconsistencies in data formats or protocols could hinder integration efforts and affect system functionality.

4. **Regulatory Compliance:** The project assumes compliance with relevant regulations and standards governing academic record-keeping, data privacy, and financial transactions. Changes to regulatory requirements or interpretations could necessitate modifications to the system architecture and functionality to ensure continued compliance.
5. **Stakeholder Involvement:** The project depends on active involvement and cooperation from stakeholders, including students, faculty members, administrative staff, and IT personnel, for requirements gathering, testing, and feedback. Limited stakeholder engagement or conflicting priorities could impact project deliverables and timelines.
6. **Availability of Development Resources:** The project assumes the availability of skilled development resources, including software engineers, testers, and project managers, to execute the development plan and deliver the system within the specified timeframe. Resource constraints or turnover could affect project progress and quality.

### **3. External Interface Requirements**

#### **3.1 User Interfaces**

1. **Degree Issuance Form**
  - a. **Description:** This interface allows students to submit degree issuance requests. It includes fields for entering personal information, academic details, and other necessary information.
2. **Complaint Form**
  - a. **Description:** This interface allows users to report spelling mistakes or errors in degree issuance documents.
3. **Dashboard for Activity Tracking:**
  - a. **Description:** This interface provides users with a dashboard to track the status of degree issuance requests.
4. **Admin Interface:**
  - a. **Description:** This interface is for administrators and departmental staff to manage degree issuance requests.
5. **Director's Dashboard:**
  - a. **Description:** This specialized interface allows the Director to monitor overall request status and departmental performance.

#### **3.2 Hardware Interfaces**

The software product interacts with hardware components of the system through various interfaces, each with specific logical and physical characteristics:

1. **Client Devices**
  - a. **Device Types:** The software supports various client devices, including desktop computers, laptops, tablets, and smartphones.

- b. Data Interaction: Client devices interact with the software to submit degree issuance requests, track request status, and receive notifications.
  - c. Control Interaction: Users control the software through input devices such as keyboards, mice, touchscreens, and styluses.
  - d. Communication Protocols: Communication between client devices and the software occurs over network protocols such as HTTP and HTTPS for web-based interactions.
- 2. Server Infrastructure
  - a. Physical Characteristics: The software is deployed on servers hosted in a data center or cloud environment.
  - b. Data Interaction: Servers manage and process degree issuance requests, store data in the database, and generate responses to client requests.
  - c. Control Interaction: Administrators control server operations through remote access tools or command-line interfaces.
  - d. Communication Protocols: Communication between client devices and server infrastructure occurs over network protocols such as TCP/IP, with data transmitted securely using encryption protocols like SSL/TLS.
- 3. Database System
  - a. Physical Characteristics: The software relies on a database management system (DBMS) hosted on dedicated servers or cloud platforms.
  - b. Data Interaction: The software interacts with the database system to store and retrieve information related to degree issuance requests, user profiles, and system configurations.
  - c. Control Interaction: Database administrators control the database system through management tools provided by the DBMS vendor.
  - d. Communication Protocols: Communication between the software and the database occurs over database-specific protocols such as MySQL and Microsoft SQL Server protocols.
- 4. Peripheral Devices
  - a. Device Types: Peripheral devices such as printers and scanners may be used for physical document processing, such as printing degree certificates or scanning supporting documents.
  - b. Data Interaction: The software may send data to peripheral devices for printing or receive data from scanners for document digitization.
  - c. Control Interaction: Users control peripheral devices through device-specific interfaces or drivers.
  - d. Communication Protocols: Communication between the software and peripheral devices may occur over standard protocols such as USB, Wi-Fi, or Bluetooth, depending on the device type.

### 3.3 Software Interfaces

The software product interfaces with various other software components to facilitate its operation and fulfill its functionality. These interfaces include connections with databases, operating systems and libraries, and integrated commercial components:

1. Database Management System (DBMS):
  - a. Name and Version: MySQL 8.0
  - b. Purpose: The software interacts with the MySQL database to store and retrieve data related to degree issuance requests, user profiles, administrative logs, and system configurations.
  - c. Data Items/Messages:
    - i. Incoming: Degree issuance requests, user authentication data, system configuration settings.
    - ii. Outgoing: Processed request data, user authentication responses, system logs.
  - d. Services Needed: CRUD operations (Create, Read, Update, Delete) for managing data records, transaction management for ensuring data integrity, and query execution for retrieving specific data sets.
  - e. Communication: The software communicates with the MySQL database using the MySQL Connector library, which provides an interface for executing SQL queries and managing database connections.
2. Operating System:
  - a. Name and Version: Windows 10
  - b. Purpose: The software runs on the Windows operating system to provide a stable and secure environment for executing application code, managing system resources, and handling network communication.
  - c. Data Items/Messages: N/A
  - d. Services Needed: Process management for executing application processes, file system management for storing application files and data, network stack for handling incoming and outgoing network requests.
  - e. Communication: The software interacts with the Windows operating system through system calls, system libraries, and system APIs provided by the Windows API.
3. .NET Framework:
  - a. Name and Version: .NET Framework 4.8
  - b. Purpose: The software is developed using the .NET framework, which provides a comprehensive set of libraries and tools for developing, deploying, and running applications on Windows platforms.
  - c. Data Items/Messages: N/A
  - d. Services Needed: Access to .NET framework libraries for implementing application logic, handling user inputs, and interacting with system resources.
  - e. Communication: The software utilizes the .NET framework's Common Language Runtime (CLR) to execute managed code and leverage .NET class libraries for various functionalities.

### 3.4 Communications Interfaces

The System will incorporate the following communication interfaces:

1. HTTP/HTTPS Protocol:

- a. Purpose: The software communicates with web browsers and clients using the Hypertext Transfer Protocol (HTTP) or its secure counterpart, HTTPS. HTTP/HTTPS is used for handling requests and responses between the frontend user interface and the backend server.
- b. Message Formatting: HTTP requests and responses are formatted according to the HTTP protocol standards, consisting of headers and optional message bodies. Requests include methods (GET, POST, PUT, DELETE) and parameters, while responses contain status codes and optional data.
- c. Communication Standards: The software adheres to the HTTP/HTTPS standards defined by the Internet Engineering Task Force (IETF) and utilizes common web communication patterns such as RESTful APIs for client-server interactions.

## 4. System Features

### 4.1 Degree Issuance Form

#### 4.1.1 Description and Priority

Students can submit a degree issuance form to initiate the process of obtaining their degree. This feature is of High priority as it is essential for students to start the degree issuance process.

#### 4.1.2 Stimulus/Response Sequences

Stimulus: Student accesses the system interface and selects the option to submit a degree issuance form.

Response: The system displays the degree issuance form with fields for personal and academic information.

Stimulus: User clicks the submit button after entering the information.

Response: The system displays a confirmation message and forwards the request to the admin.

#### 4.1.3 Functional Requirements

- ❖ REQ-1: The system shall provide a user-friendly interface for students to input necessary information for degree issuance, including program, department, student ID, and graduation details.
- ❖ REQ-2: Upon submission of the form, the system shall validate the entered information to ensure completeness and accuracy.
- ❖ REQ-3: If any errors or missing information are detected, the system shall prompt the student to correct the form before submission.
- ❖ REQ-4: After successful submission, the system shall generate a confirmation message to acknowledge receipt of the degree issuance form.
- ❖ REQ-5: The system shall log the submission date and time along with the student's details for audit and tracking purposes.
- ❖ REQ-6: In case of system errors or technical issues during form submission, the system shall display an error message and allow the student to retry the submission.

## 4.2 Complaint Form

### 4.2.1 Description and Priority

Students can submit a complaint form through the system to address issues such as spelling mistakes or inaccuracies in their degree issuance process. This feature is of Medium priority as it facilitates error reporting and ensures the accuracy of degree documents.

### 4.2.2 Stimulus/Response Sequences

Stimulus: Student accesses the system interface and selects the option to submit a complaint form.

Response: The system displays the complaint form with fields for describing the issue and attaching relevant documents if necessary.

Stimulus: User clicks the submit button after entering the information.

Response: The system displays a confirmation message and records the complaint.

### 4.2.3 Functional Requirements

- ❖ REQ-1: The system shall provide a dedicated section or form for students to submit complaints related to their degree issuance process.
- ❖ REQ-2: The complaint form shall include fields for the student to describe the issue encountered, including details such as the nature of the problem and any supporting evidence or documents.
- ❖ REQ-3: Students shall have the option to attach files or documents, such as screenshots or scanned copies, to provide additional context or evidence for their complaints.
- ❖ REQ-4: Upon submission of the complaint form, the system shall validate the entered information and uploaded documents to ensure completeness and compatibility.
- ❖ REQ-5: If any errors or missing information are detected, the system shall prompt the student to correct the form before submission.
- ❖ REQ-6: After successful submission, the system shall generate a confirmation message to acknowledge receipt of the complaint form and provide a reference number for tracking purposes.
- ❖ REQ-7: The system shall log the submission date and time along with the student's details and complaint description for audit and tracking purposes.
- ❖ REQ-8: In case of system errors or technical issues during form submission, the system shall display an error message and allow the student to retry the submission.

## 4.3 Feedback Form

### 4.3.1 Description and Priority

Students can provide feedback on their degree issuance experience through a dedicated feedback form within the system. This feature is of Medium priority as it helps gather valuable insights for improving the degree issuance process and overall user satisfaction.

#### 4.3.2 Stimulus/Response Sequences

Stimulus: Student accesses the system interface and selects the option to provide feedback.

Response: The system displays the feedback form with fields for rating the experience and providing comments or suggestions.

Stimulus: User clicks the submit button after entering the information.

Response: The system displays a confirmation message and records the feedback into the database.

#### 4.3.3 Functional Requirements

- ❖ REQ-1: The system shall include a feedback section or form where students can provide input on their degree issuance experience.
- ❖ REQ-2: The feedback form shall include fields for students to rate various aspects of the degree issuance process, such as efficiency, clarity, and responsiveness.
- ❖ REQ-3: Students shall have the option to provide additional comments or suggestions to elaborate on their ratings and provide specific feedback.
- ❖ REQ-4: Upon submission of the feedback form, the system shall validate the entered ratings and comments to ensure completeness and relevance.
- ❖ REQ-5: If any errors or missing information are detected, the system shall prompt the student to correct the form before submission.
- ❖ REQ-6: After successful submission, the system shall generate a confirmation message to acknowledge receipt of the feedback and thank the student for their input.
- ❖ REQ-7: The system shall log the submission date and time along with the student's details and feedback for analysis and improvement purposes.
- ❖ REQ-8: In case of system errors or technical issues during form submission, the system shall display an error message and allow the student to retry the submission.

### 4.4 Activity Tracking for Students

#### 4.4.1 Description and Priority

Students shall have access to a dashboard within the system to monitor the status of their degree issuance requests, providing updates on pending, processing, and delivered requests. This feature is of Medium priority as it enhances transparency and provides students with real-time visibility into the progress of their requests.

#### 4.4.2 Stimulus/Response Sequences

Stimulus: Student logs into the system and navigates to the dashboard section.

Response: The system displays the dashboard with details of the student's degree issuance requests, including their current status.

#### 4.4.3 Functional Requirements

- ❖ REQ-1: The system shall provide a dashboard interface accessible to students, displaying the status of their degree issuance requests.
- ❖ REQ-2: The dashboard shall include separate sections or tabs for pending, processing, and delivered requests, categorizing them based on their status.



- ❖ REQ-3: For each degree issuance request, the dashboard shall display relevant details such as request ID, submission date, current status, and any updates or comments provided by the FYP department or Finance department.
- ❖ REQ-4: The system shall update the dashboard in real-time to reflect changes in the status of degree issuance requests, ensuring that students have access to the latest information.
- ❖ REQ-5: Students shall have the ability to filter and search for specific requests within the dashboard based on criteria such as request ID or status.
- ❖ REQ-6: The system shall provide options for students to view additional details or history related to their degree issuance requests, including any previous actions taken or communications received.
- ❖ REQ-7: If there are any updates or changes to the status of a degree issuance request, the system shall notify the student through email or in-app notifications to ensure timely awareness.
- ❖ REQ-8: The dashboard interface shall be intuitive and user-friendly, allowing students to easily navigate and understand the information presented.

## 4.5 Degree Receipt for Students

### 4.5.1 Description and Priority

Upon completion of the degree issuance process, students shall receive their degrees either electronically or physically. This feature is of High priority as it signifies the culmination of the degree issuance process and the attainment of the student's academic qualification.

### 4.5.2 Stimulus/Response Sequences

Stimulus: Degree issuance process is completed successfully for a student.

Response: The system generates and delivers the degree to the student, either electronically or physically, based on their preference..

### 4.5.3 Functional Requirements

- ❖ REQ-1: The system shall provide options for students to choose their preferred method of receiving their degree, either electronically (e.g., PDF download) or physically (e.g., mail delivery).
- ❖ REQ-2: If the student chooses electronic delivery, the system shall generate the degree document in a secure format (e.g., PDF) and make it available for download through the student's account.
- ❖ REQ-3: If the student chooses physical delivery, the system shall initiate the printing and mailing process to send the degree document to the student's registered mailing address.
- ❖ REQ-4: The system shall ensure the accuracy and integrity of the degree document, including verifying the student's information and academic credentials before issuance.
- ❖ REQ-5: Upon successful generation or delivery of the degree document, the system shall update the status of the degree issuance request to indicate that the degree has been received by the student.

- ❖ REQ-6: If there are any issues or delays in generating or delivering the degree document, the system shall notify the student and provide updates on the resolution process.
- ❖ REQ-7: The system shall maintain a record of degree issuance and delivery for auditing and tracking purposes, including details such as issuance date, delivery method, and student confirmation.

## **4.6 FYP Department - Student Notification and Decision Making**

### **4.6.1 Description and Priority**

The system shall notify the FYP department about degree issuance requests submitted by students and enable them to make decisions (accept, reject, or raise objections) with comments. This feature is of High priority as it directly impacts the processing and approval of degree issuance requests.

### **4.6.2 Stimulus/Response Sequences**

Stimulus: The system receives a degree issuance request from a student.

Response: The system automatically sends a notification to the FYP department regarding the new request, prompting them to review and provide a decision.

### **4.6.3 Functional Requirements**

- ❖ REQ-1: The system shall automatically generate and send notifications to the FYP department upon the submission of a degree issuance request by a student.
- ❖ REQ-2: The notification to the FYP department shall include relevant details of the degree issuance request, such as student information and request details.
- ❖ REQ-3: The FYP department shall have the ability to access the degree issuance request details, including the degree issuance form submitted by the student.
- ❖ REQ-4: The system shall provide options for the FYP department to make decisions on the degree issuance request, including acceptance, rejection, or raising objections.
- ❖ REQ-5: When making a decision, the FYP department shall have the option to add comments or explanations to provide context for their decision.
- ❖ REQ-6: The system shall update the status of the degree issuance request based on the decision provided by the FYP department.
- ❖ REQ-7: If the FYP department raises objections to the degree issuance request, the system shall notify the relevant parties (e.g., student, One Stop Admin) and provide details of the objections for resolution.
- ❖ REQ-8: The system shall log all interactions and decisions made by the FYP department for audit and tracking purposes.

## **4.7 Finance Department - Student Notification and Payment Verification**

### **4.7.1 Description and Priority**

The system shall notify the Finance department about degree issuance requests submitted by students and enable them to verify outstanding amounts and degree

issuance fees before making decisions. This feature is of High priority as it ensures financial compliance and accuracy in the degree issuance process.

#### 4.7.2 Stimulus/Response Sequences

Stimulus: Degree issuance request submitted by a student.

Response: The system automatically sends a notification to the Finance department, prompting them to review and verify payment details.

#### 4.7.3 Functional Requirements

- ❖ REQ-1: The system shall automatically generate and send notifications to the Finance department upon the submission of a degree issuance request by a student.
- ❖ REQ-2: The notification to the Finance department shall include relevant details of the degree issuance request, such as student information and payment status.
- ❖ REQ-3: The Finance department shall have access to the degree issuance request details, including the degree issuance form and payment history.
- ❖ REQ-4: The system shall display information regarding outstanding amounts and degree issuance fees associated with the degree issuance request for verification by the Finance department.
- ❖ REQ-5: The Finance department shall have the ability to verify the payment status and fee payments related to the degree issuance request.
- ❖ REQ-6: If any discrepancies or outstanding amounts are detected, the Finance department shall have the option to communicate with the student or relevant stakeholders for resolution.
- ❖ REQ-7: Upon verification of payment details, the Finance department shall provide input or decisions on the degree issuance request, including acceptance, rejection, or raising objections.
- ❖ REQ-8: The system shall log all interactions and decisions made by the Finance department for audit and tracking purposes.

### 4.8 Decision Making for Finance Department

#### 4.8.1 Description and Priority

The Finance department shall have the capability to provide decisions (accept, reject, or raise objections) for degree issuance requests submitted by students, along with comments. This feature is of High priority as it directly influences the approval and processing of degree issuance requests based on financial considerations.

#### 4.8.2 Stimulus/Response Sequences

Stimulus: Degree issuance request details displayed to the Finance department.

Response: The Finance department provides a decision (accept, reject, or raise objections) along with comments for the degree issuance request.

#### 4.8.3 Functional Requirements

- ❖ REQ-1: The system shall provide an interface for the Finance department to access degree issuance requests awaiting their decision.
- ❖ REQ-2: The Finance department shall have options to provide decisions on degree issuance requests, including acceptance, rejection, or raising objections.
- ❖ REQ-3: When providing a decision, the Finance department shall have the ability to add comments or explanations to provide context for their decision.
- ❖ REQ-4: The system shall update the status of the degree issuance request based on the decision provided by the Finance department.
- ❖ REQ-5: If the Finance department raises objections to the degree issuance request, the system shall notify the relevant parties (e.g., student, One Stop Admin) and provide details of the objections for resolution.
- ❖ REQ-6: The system shall log all decisions made by the Finance department, along with any associated comments, for audit and tracking purposes.
- ❖ REQ-7: In case of system errors or technical issues affecting decision-making functionality, the system shall log and report the incident for resolution.

## **4.9 Student Notification Management and Token Generation for One Stop Admin**

### **4.9.1 Description and Priority**

The One Stop Admin shall manage notifications of degree issuance requests and generate tokens for each request, indicating the time required for degree issuance. This feature is of High priority as it facilitates the efficient handling and tracking of degree issuance requests by the admin.

### **4.9.2 Stimulus/Response Sequences**

Stimulus: Degree issuance request submitted by a student.

Response: The system sends a notification to the One Stop Admin, prompting them to generate a token for the request.

### **4.9.3 Functional Requirements**

- ❖ REQ-1: The system shall automatically send notifications to the One Stop Admin upon the submission of a degree issuance request by a student.
- ❖ REQ-2: The notification to the One Stop Admin shall include relevant details of the degree issuance request, such as student information and request details.
- ❖ REQ-3: The One Stop Admin shall have access to a dashboard or interface displaying degree issuance requests awaiting their attention.
- ❖ REQ-4: The One Stop Admin shall be able to generate a unique token for each degree issuance request, indicating the time required for degree issuance.
- ❖ REQ-5: The generated token shall contain information such as the request ID, submission date, and estimated time for completion.
- ❖ REQ-6: The system shall update the status of the degree issuance request to indicate that a token has been generated by the One Stop Admin.

- ❖ REQ-7: If there are any issues or delays in generating tokens, the system shall notify the One Stop Admin and provide assistance for resolution.
- ❖ REQ-8: The system shall log all token generation activities performed by the One Stop Admin for audit and tracking purposes.

## **4.10 Request Management Dashboard for One Stop Admin**

### **4.10.1 Description and Priority**

The One Stop Admin shall have access to a dashboard for managing degree issuance requests, including viewing all pending, processed, and new requests, and addressing objections raised by the FYP or Finance departments. This feature is of High priority as it centralizes request management and ensures timely resolution of issues.

### **4.10.2 Stimulus/Response Sequences**

Stimulus: One Stop Admin logs into the system and navigates to the dashboard section.

Response: The system displays the dashboard with categorized lists of degree issuance requests and notifications of objections raised by FYP or Finance departments.

### **4.10.3 Functional Requirements**

- ❖ REQ-1: The system shall provide a dashboard interface accessible to the One Stop Admin, categorizing degree issuance requests into pending, processed, and new requests.
- ❖ REQ-2: The dashboard shall display detailed information for each degree issuance request, including student details, request status, and any objections raised by the FYP or Finance departments.
- ❖ REQ-3: The One Stop Admin shall have the ability to filter and search for specific requests within the dashboard based on criteria such as request ID, student name, or status.
- ❖ REQ-4: The system shall notify the One Stop Admin of any objections raised by the FYP or Finance departments regarding degree issuance requests, providing relevant details for review.
- ❖ REQ-5: The One Stop Admin shall be able to review objections raised by the FYP or Finance departments and take necessary actions to address them, such as communicating with relevant stakeholders or initiating resolution processes.
- ❖ REQ-6: The system shall update the status of degree issuance requests and objections in real-time within the dashboard, ensuring that the One Stop Admin has access to the latest information.
- ❖ REQ-7: If there are any errors or technical issues affecting the dashboard functionality, the system shall log and report the incident for resolution.

## 4.11 Degree Issuance and Transcript Generation for One Stop Admin

### 4.11.1 Description and Priority

The One Stop Admin shall have the authority to issue degrees to students, update the processed request list with timestamps, and generate transcripts containing relevant academic information. This feature is of High priority as it represents the final stage of the degree issuance process and ensures accurate record-keeping of academic achievements.

### 4.11.2 Stimulus/Response Sequences

Stimulus: One Stop Admin accesses the system interface to process degree issuance requests.

Response: The system provides options for degree issuance, updates request lists, and generates transcripts upon admin action.

### 4.11.3 Functional Requirements

- ❖ REQ-1: The system shall provide an interface for the One Stop Admin to review and process degree issuance requests.
- ❖ REQ-2: The One Stop Admin shall be able to issue degrees to students upon successful completion of the degree issuance process.
- ❖ REQ-3: Upon issuing a degree, the One Stop Admin shall update the processed request list with the current timestamp, indicating the completion time.
- ❖ REQ-4: The system shall generate transcripts for each student upon degree issuance, containing relevant academic information such as department, CGPA, course grades, duration of study, and photograph.
- ❖ REQ-5: Transcripts generated by the system shall be accurate and reflect the student's academic achievements based on the degree issuance information.
- ❖ REQ-6: The One Stop Admin shall have the ability to review and verify transcripts before issuing them to students to ensure accuracy and completeness.
- ❖ REQ-7: The system shall maintain a record of issued degrees and generated transcripts for audit and tracking purposes, including details such as issuance date, student information, and transcript content.
- ❖ REQ-8: If there are any errors or discrepancies in the degree issuance process or transcript generation, the system shall notify the One Stop Admin and provide options for resolution.

## 4.12 Request Monitoring and Activity Tracking Dashboard for Director

### 4.12.1 Description and Priority

The Director shall be equipped with a dashboard enabling the comprehensive monitoring of degree issuance requests, including viewing all requests generated per day, pending requests, processed requests, time taken by each department, and

activity tracking logs. This feature is of High priority as it facilitates oversight, decision-making, and optimization of the degree issuance process.

#### 4.12.2 Stimulus/Response Sequences

Stimulus: The Director accesses the system interface and navigates to the dashboard section.

Response: The system presents a dashboard displaying detailed metrics and logs related to degree issuance requests and activities.

#### 4.12.3 Functional Requirements

- ❖ REQ-1: The system shall provide a dashboard interface accessible to the Director, displaying metrics and logs relevant to degree issuance requests and activities.
- ❖ REQ-2: The dashboard shall include sections for viewing all requests generated per day, categorizing them by date.
- ❖ REQ-3: The Director shall have the ability to view lists of pending and processed requests, categorized based on their status.
- ❖ REQ-4: The system shall display the time taken by each department (e.g., FYP, Finance) to process degree issuance requests, enabling performance analysis and optimization.
- ❖ REQ-5: Activity tracking logs shall be accessible to the Director, providing details of key activities such as request processing stages, notifications, and decision-making.
- ❖ REQ-6: The system shall update the dashboard in real-time to reflect the latest metrics and logs related to degree issuance requests and activities.
- ❖ REQ-7: The Director shall have the option to filter and search for specific information within the dashboard based on criteria such as date, department, or request status.
- ❖ REQ-8: If there are any errors or technical issues affecting the dashboard functionality, the system shall log and report the incident for resolution.

## 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

#### 5.1.1 Response Time

- ❖ The system should respond to user interactions, such as form submissions and dashboard navigation, within 2 seconds under normal load conditions.
- ❖ Degree issuance processes, including generating tokens and issuing degrees, should be completed within 5 minutes for each request.

#### 5.1.2 Scalability

- ❖ The system should be capable of handling a minimum of 100 simultaneous degree issuance requests without significant degradation in performance.

- ❖ It should accommodate future growth by scaling to support up to 500 simultaneous requests with minimal impact on response times.

#### 5.1.3 Database Performance

- ❖ Database queries for retrieving degree issuance request details and activity logs should execute within 1 second on average.
- ❖ The database should be optimized to handle large volumes of data efficiently, ensuring smooth operation even with extensive record-keeping.

#### 5.1.4 Dashboard Load Time

- ❖ The dashboard interface for administrators and directors should load within 3 seconds, providing quick access to request monitoring and activity tracking features.
- ❖ The load time should remain consistent even as the number of requests and activity logs increases over time.

#### 5.1.5 Concurrent User Handling

- ❖ The system should support a minimum of 50 concurrent users accessing different modules simultaneously without experiencing performance bottlenecks.
- ❖ It should be capable of gracefully handling peak loads, such as during registration periods or end-of-semester rushes, without affecting overall system responsiveness.

#### 5.1.6 Transaction Throughput

- ❖ The system should support a minimum of 200 transactions per minute, including form submissions, notifications, and database updates.
- ❖ Transaction throughput should be monitored and optimized to ensure efficient use of system resources and maintain acceptable performance levels.

## 5.2 Safety Requirements

#### 5.2.1 Data Security

- ❖ All sensitive student information, including personal details and academic records, should be encrypted both in transit and at rest to prevent unauthorized access.
- ❖ Access controls should be implemented to ensure that only authorized personnel can view or modify sensitive data, with role-based permissions enforced.
- ❖ Regular security audits and vulnerability assessments should be conducted to identify and address potential security risks proactively.

#### 5.2.2 Backup and Recovery

- ❖ The system should regularly back up all data, including degree issuance requests, activity logs, and user information, to prevent data loss in the event of system failure or corruption.
- ❖ Backup procedures should be automated and scheduled to occur at least once daily, with backup copies stored securely off-site or in a separate location.
- ❖ In case of data loss or system failure, the system should have mechanisms in place to facilitate timely data recovery and restoration to minimize disruption to operations.

#### 5.2.3 Audit Trails



- ❖ The system should maintain comprehensive audit trails of all user interactions, including login attempts, form submissions, and administrative actions, to ensure accountability and traceability.
- ❖ Audit logs should include timestamps, user identifiers, and details of the actions performed, with sufficient granularity to reconstruct user activities and system events accurately.
- ❖ Audit trails should be tamper-evident, with measures in place to detect and prevent unauthorized modifications or deletions of log data.

#### 5.2.4 System Reliability

- ❖ The system should be highly reliable, with minimal downtime and service disruptions to ensure continuous availability for users.
- ❖ Redundant infrastructure components, such as servers, databases, and network connections, should be deployed to mitigate the impact of hardware failures or maintenance activities.
- ❖ Proactive monitoring and alerting mechanisms should be in place to detect potential issues and address them promptly to maintain system reliability.

#### 5.2.5 User Training and Awareness

- ❖ Users, including administrators and staff members, should receive comprehensive training on system usage, security best practices, and data handling procedures to promote safe and responsible use of the system.
- ❖ Regular security awareness sessions and updates should be conducted to keep users informed about emerging threats and reinforce security protocols and policies.

### 5.3 Security Requirements

#### 5.3.1 Authentication and Authorization

- ❖ Users should be required to authenticate themselves using strong, multi-factor authentication mechanisms, such as username/password combinations combined with SMS verification codes or biometric authentication.
- ❖ Role-based access control (RBAC) should be enforced, ensuring that users only have access to the functionalities and data relevant to their roles and responsibilities.

#### 5.3.2 Data Encryption

- ❖ All sensitive data, including student records, financial information, and administrative logs, should be encrypted using industry-standard encryption algorithms during transmission over networks and storage in databases.
- ❖ Encryption keys should be securely managed and stored separately from the encrypted data to prevent unauthorized access.

#### 5.3.3 Secure Communication

- ❖ Secure communication protocols, such as HTTPS, should be used to encrypt data exchanged between clients and the server, preventing eavesdropping and tampering by malicious actors.

- ❖ Data validation and sanitization should be performed on user inputs to mitigate the risk of injection attacks, such as SQL injection and cross-site scripting (XSS).

#### 5.3.4 Access Control

- ❖ Access controls should be implemented to restrict access to sensitive functionalities and data, ensuring that only authorized users can perform actions such as issuing degrees, modifying requests, and viewing confidential information.
- ❖ Access control lists (ACLs) should be enforced at the application level, with granular permissions assigned based on user roles and responsibilities.

#### 5.3.5 Audit Logging

- ❖ The system should maintain detailed audit logs of all user activities, system events, and security-related incidents, including login attempts, data accesses, and configuration changes.
- ❖ Audit logs should be tamper-evident, with strict controls in place to prevent unauthorized modifications or deletions, ensuring their integrity and reliability for forensic analysis.

#### 5.3.6 Security Monitoring

- ❖ Continuous security monitoring should be implemented to detect and respond to potential security threats and anomalies in real-time, such as unauthorized access attempts, suspicious login patterns, and unusual data access activities.
- ❖ Intrusion detection and prevention systems (IDPS) should be deployed to monitor network traffic and identify malicious activities, with automated alerts triggered for immediate response by security personnel.

## 5.4 Software Quality Requirements

### 5.4.1 Adaptability

- ❖ The system should be adaptable to evolving requirements and changing environments, allowing for seamless integration of new features and functionalities as needed.
- ❖ Verifiable: The system should support plugin architectures or modular components that can be easily extended or modified without disrupting existing functionality.

### 5.4.2 Availability

- ❖ The system should ensure high availability, with uptime exceeding 99% during peak usage hours, to minimize service disruptions and ensure continuous access for users.
- ❖ Verifiable: Regular uptime monitoring and reporting should be conducted, with service level agreements (SLAs) established to define availability targets and penalties for downtime.

### 5.4.3 Correctness

- ❖ The system should prioritize correctness, ensuring accurate processing of degree issuance requests and adherence to business rules and regulatory requirements.

- ❖ Verifiable: Automated testing suites should be developed to verify the correctness of system functionalities, with code reviews and quality assurance processes in place to identify and address defects.

#### 5.4.4 Flexibility

- ❖ The system should exhibit flexibility, allowing for customization and configuration to meet the unique needs of different institutions or departments.
- ❖ Verifiable: Configuration options and parameters should be well-documented and easily adjustable, enabling administrators to tailor the system to their specific requirements without requiring code changes.

#### 5.3.5 Interoperability

- ❖ The system should support interoperability with external systems and services, facilitating data exchange and integration with third-party applications or platforms.
- ❖ Verifiable: Compliance with industry standards and protocols for data interchange, such as RESTful APIs or standardized data formats, should be demonstrated through interoperability testing and certification.

#### 5.4.6 Portability

- ❖ The system should be portable across different hardware and software environments, enabling deployment on various operating systems, databases, and hosting platforms.
- ❖ Verifiable: Compatibility testing should be performed to validate the system's functionality and performance on different platforms, with documentation provided for supported configurations and dependencies.

#### 5.4.7 Reusability

- ❖ The system should promote reusability of components and modules, allowing for efficient development and maintenance of future projects.
- ❖ Verifiable: Code repositories and libraries should be organized and documented to facilitate code reuse, with metrics tracking the percentage of reusable code across projects.

#### 5.4.8 Robustness

- ❖ The system should be robust, capable of handling unexpected inputs, errors, and edge cases gracefully without crashing or compromising data integrity.
- ❖ Verifiable: Stress testing and fault injection should be conducted to assess the system's resilience under adverse conditions, with recovery mechanisms in place to restore normal operation after failures.

#### 5.4.9 Testability

- ❖ The system should be designed for testability, with modular architecture and comprehensive test suites enabling thorough validation of functionalities and behaviors.
- ❖ Verifiable: Test coverage metrics should be tracked, with automated testing frameworks and continuous integration pipelines ensuring that all code changes are rigorously tested before deployment.

#### 5.4.10 Usability

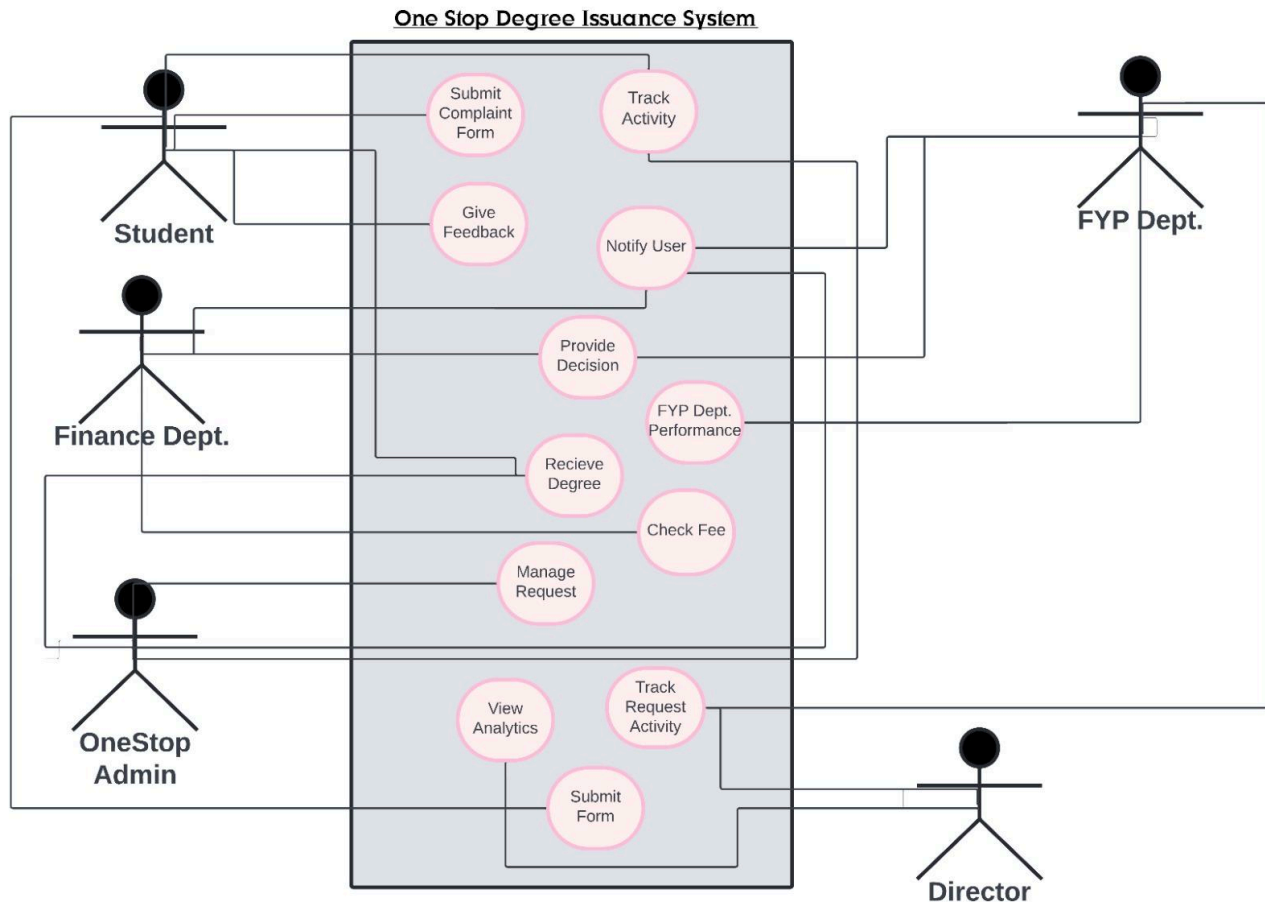
- ❖ The system should prioritize usability, providing an intuitive and user-friendly interface that minimizes the learning curve and enhances user satisfaction.
- ❖ Verifiable: Usability testing sessions should be conducted with representative users to gather feedback and identify areas for improvement, with metrics such as task completion rates and user satisfaction scores used to evaluate usability enhancements.

## 5.5 Business Rules

1. Only students who have fulfilled all academic requirements and cleared any outstanding dues are eligible to submit degree issuance requests.
2. The FYP department is responsible for reviewing and approving degree issuance requests related to final year projects, ensuring compliance with project guidelines and academic standards.
3. The Finance department must verify that students have paid all outstanding fees and the degree issuance fee before approving degree issuance requests.
4. Degree issuance requests must be processed within a specified timeframe, and any delays or objections raised by the FYP or Finance departments should be communicated to the students promptly.
5. The One Stop Admin is responsible for coordinating the degree issuance process, generating tokens for requests, and ensuring timely communication with students and relevant departments.
6. The Director has oversight of the degree issuance process and is responsible for monitoring request status, departmental performance, and overall activity tracking.
7. All degree issuance activities should be logged for audit and tracking purposes, including timestamps, user identifiers, and relevant details of the actions performed.
8. The system should enforce role-based access control to restrict access to sensitive functionalities and data based on user roles and permissions.
9. Requests for degree issuance should follow a predefined workflow, with sequential reviews and approvals by the FYP, Finance, and One Stop Admin departments before issuance.
10. Any changes to degree issuance policies or procedures should be documented and communicated to all relevant stakeholders in a timely manner.

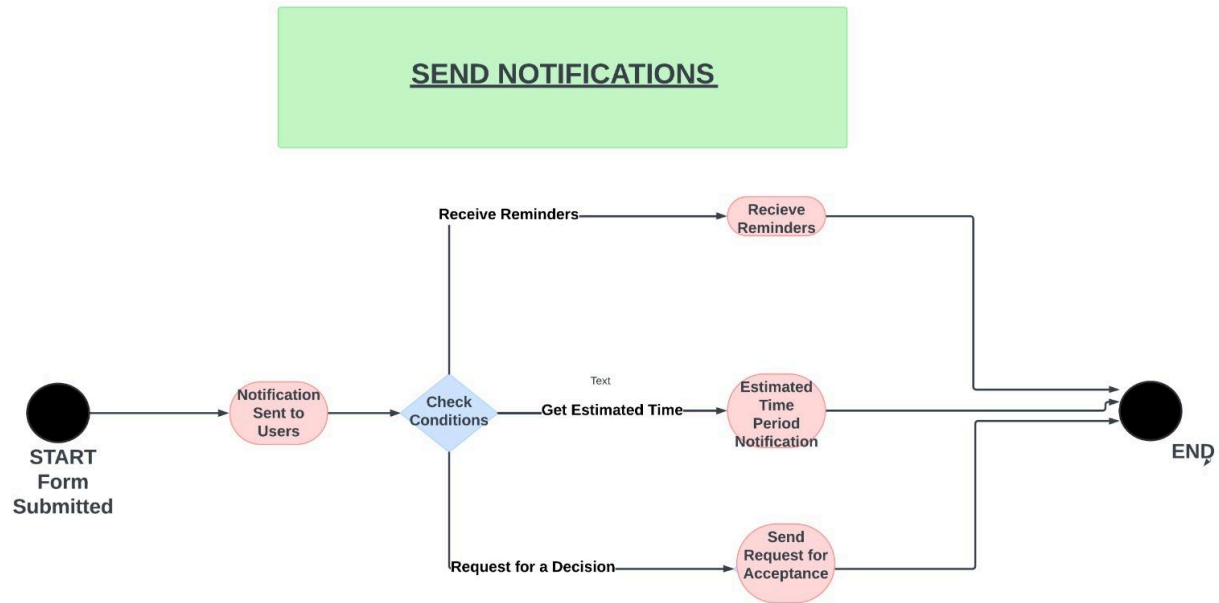
## 6. Diagrams

## 6.1 Use Case Diagram

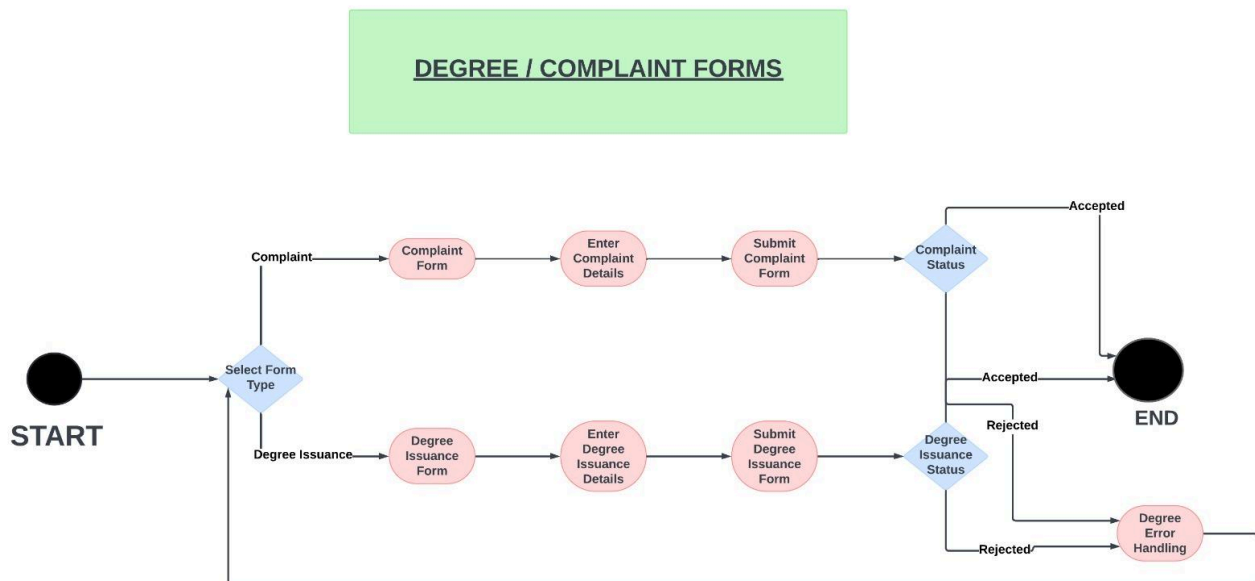


## 6.2 Activity Diagram

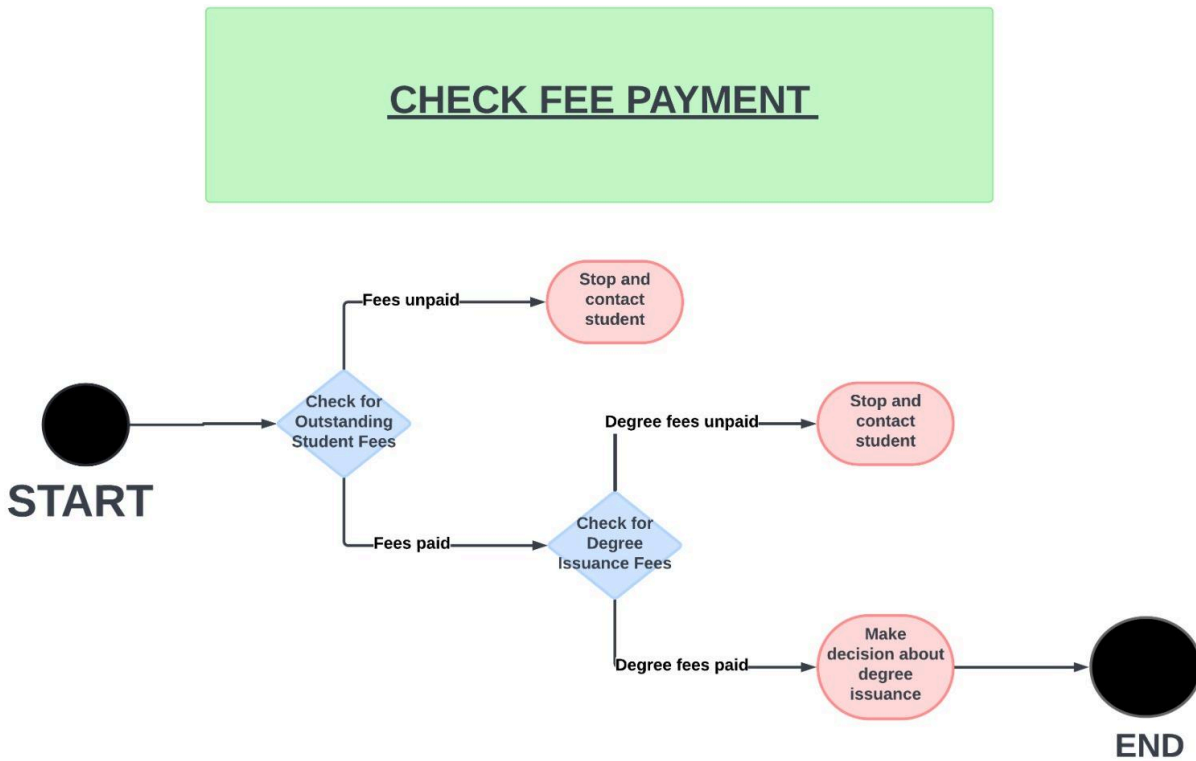
### 6.2.1 Notification Module



### 6.2.2 Degree Issuance and COMplaint Form

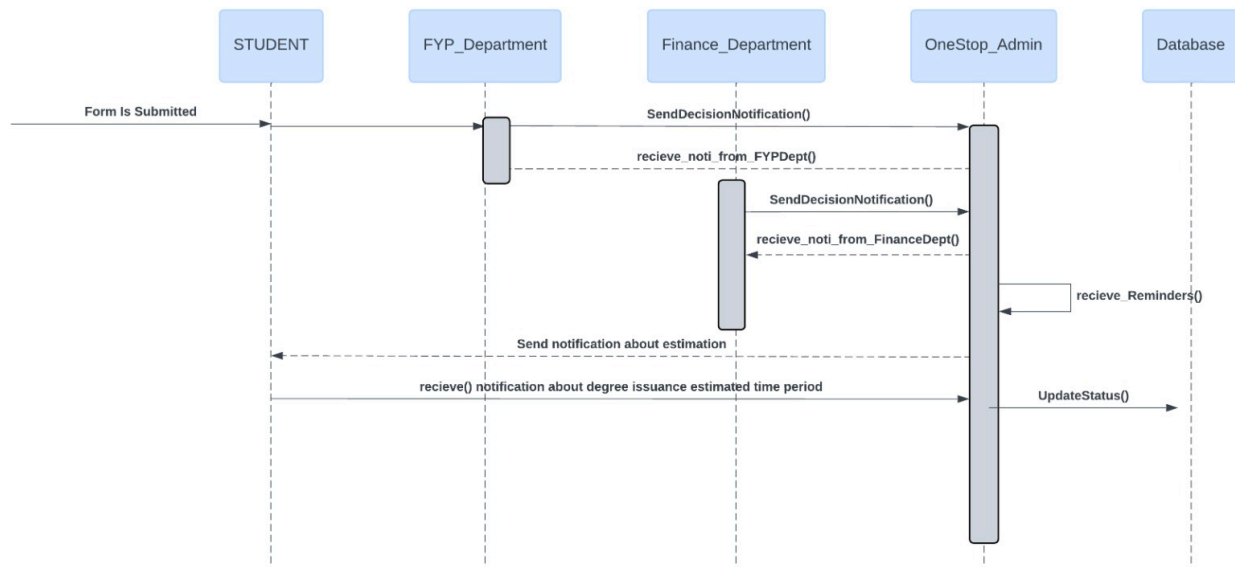


### 6.2.3 Finance Department Decision Making and Payment Verification

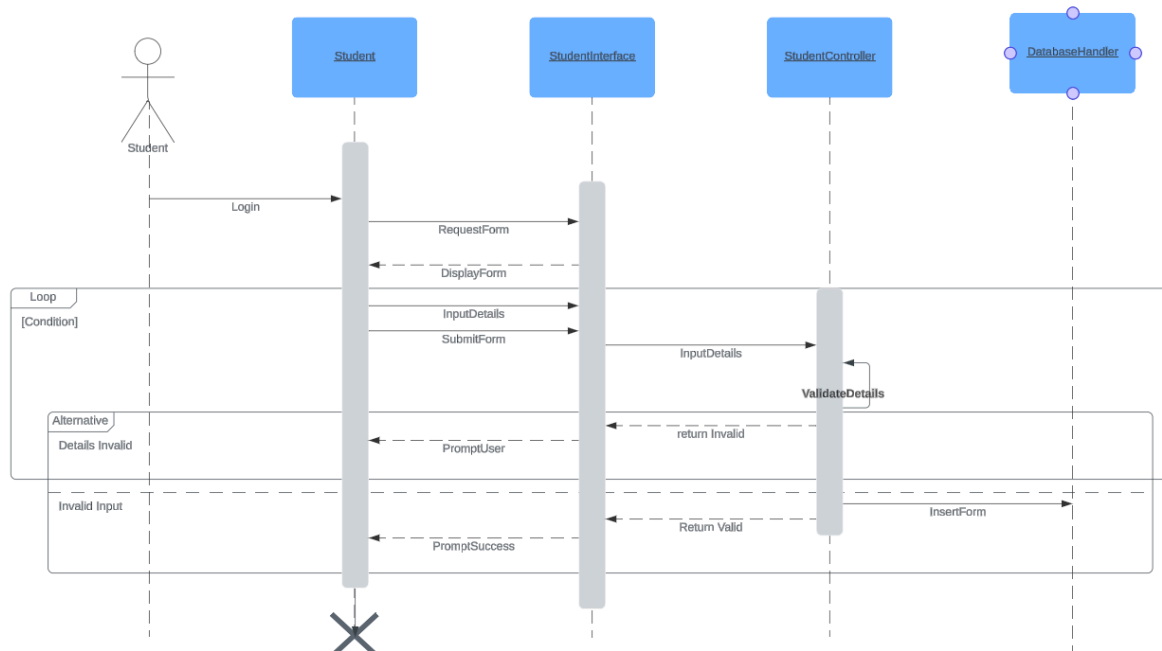


## 6.3 Sequence Diagram

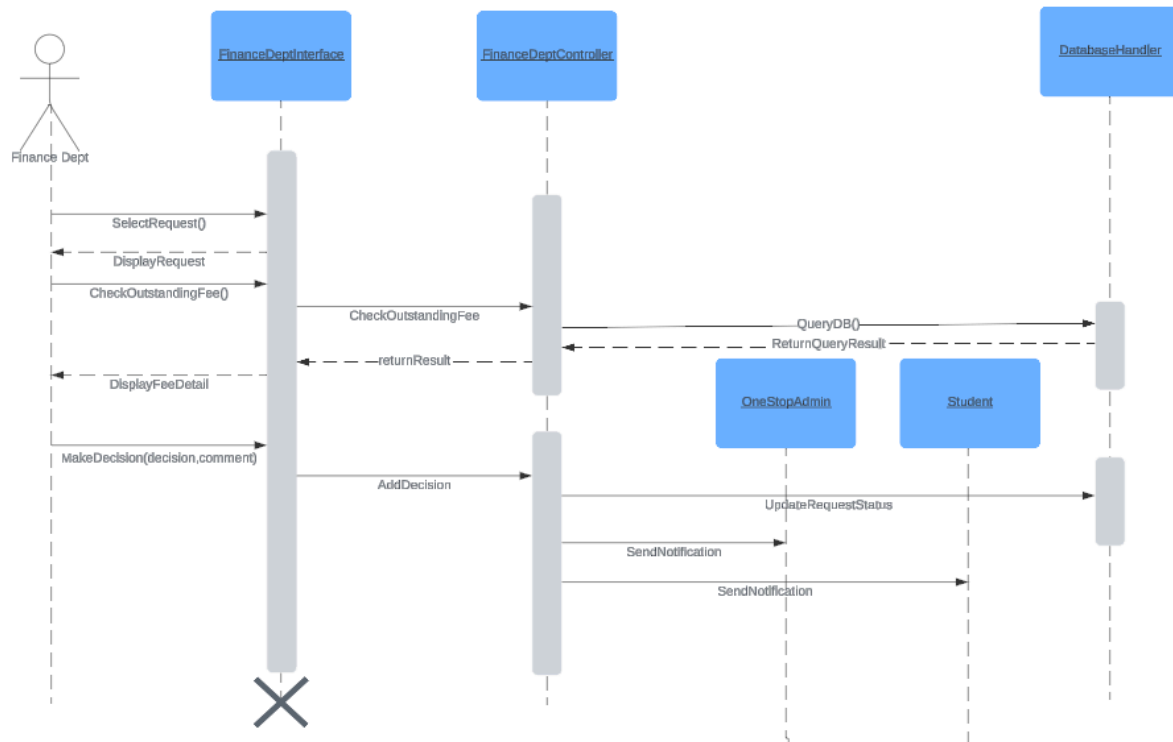
### 6.3.1 Notification Module



### 6.3.2 Degree Issuance Form



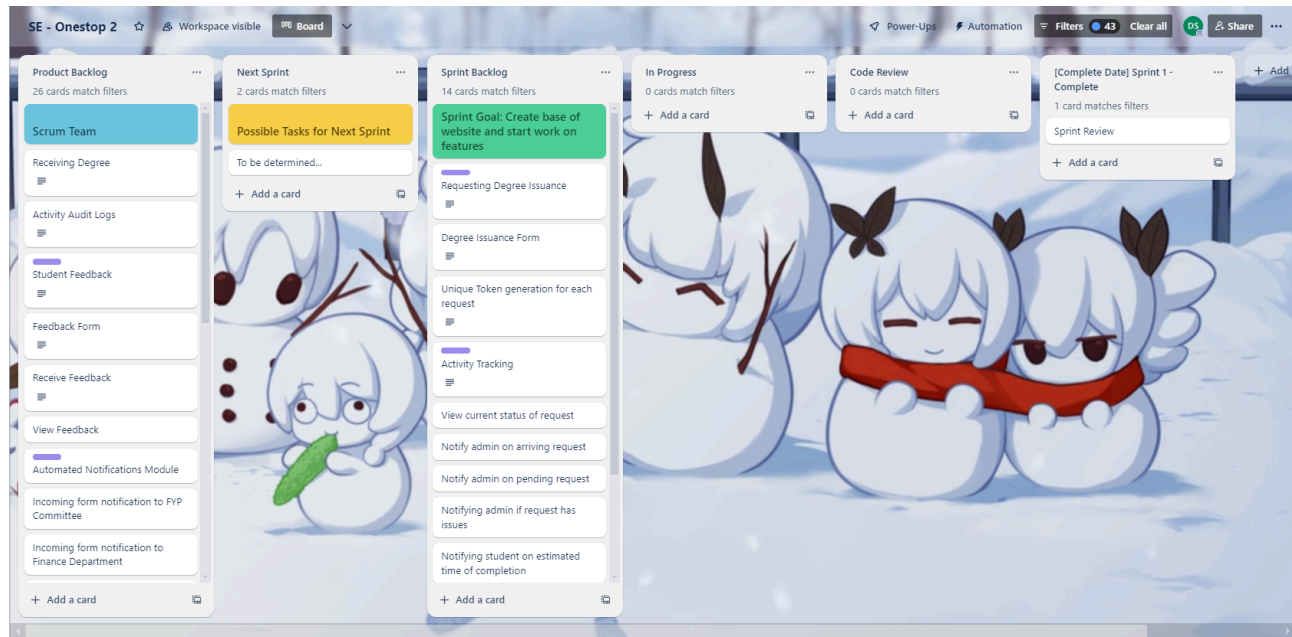
### 6.3.3 Finance Department Decision Making and Payment Verification



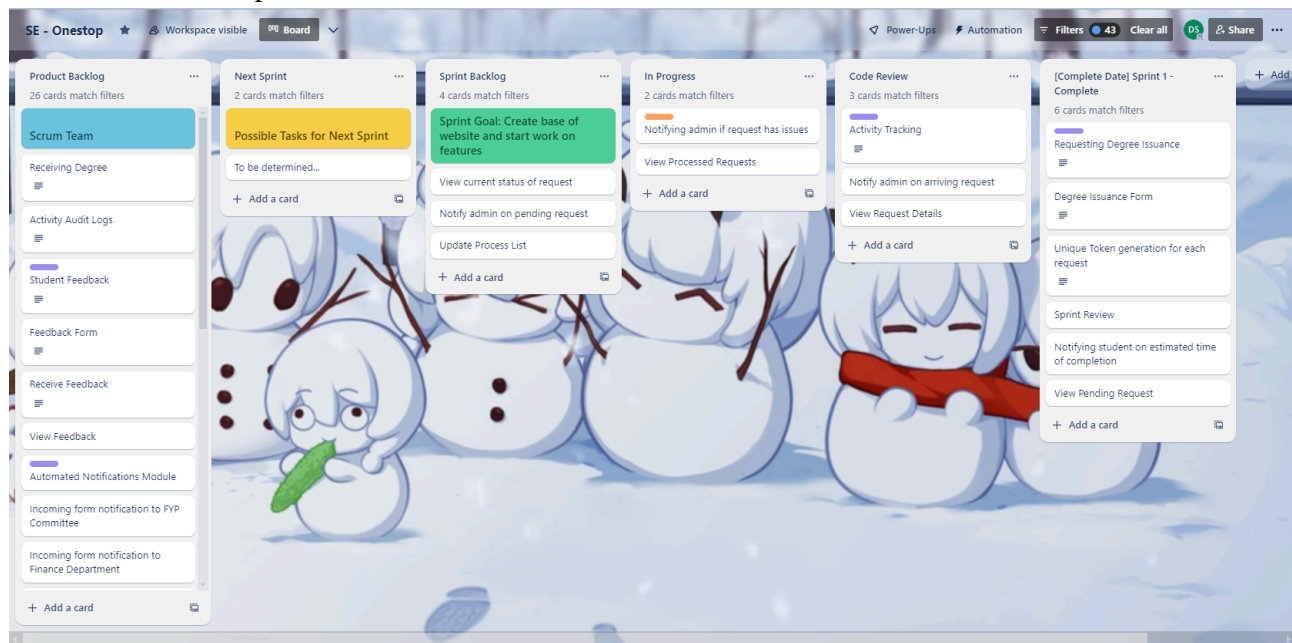


## 7. Trello Board

### 7.1 Start of Sprint



### 7.2 Middle of Sprint



### 7.3 End of Sprint



## 7.4 Scrum Master Actions

Sprint Review

in list [7/01/24 Sprint 1 - Complete](#)

Labels

Expedite

Blocked

Bug

Main Feature

+

Notifications

👁 Watch

Add to card

Members

Labels

Checklist

Dates

Attachment

Custom Fields

Description

Edit

Sprint Review for Sprint 1  
Start Date: 1/04/24  
  
End Date: 7/04/24

Activity

Show details

DS

Write a comment...

IM

i210822 Abdullah Masood Mughal

3 minutes ago

Thanks for sharing, team. It's important to gather feedback from users to continuously improve our product. Let's continue with the rest of our user stories and address any questions or concerns from stakeholders.

🗨

Reply

Delete

DS

Daniyal Kaleem Sheikh

3 minutes ago

@i210822abdullahmasoodmughal We received a few suggestions for improving the user interface of the complaint form. We'll take those into account for our next sprint planning session.

🗨

Edit

Delete

IM

i210822 Abdullah Masood Mughal

3 minutes ago

@muhammadtayyabsohail Nice work! Did we receive any feedback or suggestions from users during the sprint that we should consider for future iterations?

🗨

Reply

Delete

MS

Muhammad Tayyab Sohail

4 minutes ago

@i210822abdullahmasoodmughal That was me. We implemented the complaint form functionality, allowing students to report any spelling mistakes or issues with their degree issuance requests.

## 8. Github

## Appendix A: Glossary

1. **FYP:** Final Year Project.
2. **One Stop Admin:** Administrator responsible for managing the degree issuance process and coordinating activities among different departments.
3. **Director:** The individual overseeing the degree issuance process and responsible for monitoring request status and departmental performance.
4. **Finance Department:** Department responsible for handling financial matters related to degree issuance, including fee verification and payment processing.
5. **FYP Department:** Department responsible for reviewing and approving degree issuance requests related to final year projects.
6. **User:** Individual interacting with the system, including students, administrators, and departmental staff.
7. **Dashboard:** A visual interface that displays key metrics, data, and information relevant to the user's role or responsibilities.
8. **Audit Logs:** Detailed records of user activities, system events, and security-related incidents for accountability and traceability purposes.
9. **Token:** A unique identifier or code assigned to each degree issuance request for tracking and reference purposes.
10. **Interoperability:** The ability of the system to interact and exchange data with external systems or platforms.