Information Systems Department

IS499 Senior Project - semester 241

Dr. Suliman Mohamed Fati & Dr. Bayan AlGhofaily

**Table of Contents**

[**Abstract 4**](#_tm5n5v11y5mq)

[**Introduction 5**](#_n74g0oi4n3u4)

[Background and Motivation 5](#_sn14zrx3i9yf)

[Problem Statement 6](#_8v84bl1lu70x)

[Objectives of Project 6](#_acljj7qmu0a)

[Scope of the Project 7](#_8f43pjqreb1)

[Project Relevance and Importance 8](#_fesh15ue6k45)

[Project Constraints and Assumptions 8](#_z10vhaqrpejc)

[**Literature Review 9**](#_jhunfimojkoc)

[Academic Workflow Systems in Higher Education 9](#_s2scz7du5zn)

[Use Cases in Registry Management 9](#_1qrth2zzq4o)

[Comparative Analysis of Traditional vs Digital Workflows 10](#_r7avj6s5tl8j)

[Laravel Framework in Educational Platforms 10](#_aga674ugu9tt)

[Role-Based Access Control (RBAC) Systems 11](#_7jd2n4d2cilh)

[Security in Web Applications 12](#_39wfkcrhnqs)

[MySQL for Academic Data Handling 12](#_i8apul5sfb03)

[Responsive Web UI Trends for Student Portals 13](#_61qqhcmwon09)

[**System Design and Architecture 13**](#_3drcotevcryh)

[Overview 14](#_saycyt9zfar3)

[System Architecture Diagram 14](#_oixd1uecuvq7)

[Database Schema 15](#_3kzw60jrtrln)

[**Functional Modules 16**](#_dgrse1ahbqq3)

[User Management Module 16](#_lmqcu7vhsryn)

[Academic Vow Module 17](#_8nnycyg5ycpq)

[Registry Excess Management Module 17](#_5gtnbj4dr6gm)

[Notification and Messaging System 18](#_gwan28nyom71)

[Audit Logs and Activity Tracking 18](#_d15emgz8bznw)

[**Software Design 18**](#_ecdntc5o1n1u)

[Laravel Folder Structure Overview 19](#_kldbuinxzz7d)

[Web Route Configuration and Access Logic 19](#_w75t496ul5yc)

[User Interface Design with Blade Templates 20](#_4jglg5v92c89)

[Bootstrap and Custom CSS Responsive Design 20](#_qqdiyo5gskjj)

[PHP and Laravel 20](#_fhkuw1ubm519)

[**System Workflows and Diagrams 21**](#_sl4e4wasjjrg)

[Sequence Diagram 21](#_3789okgnq8nn)

[Use Case Diagram 22](#_mjs9038s5l81)

[Activity Diagram 23](#_n2s3kn5yhmu)

[**System Implementation 24**](#_brghifyj27f4)

[Installation and Deployment Steps 24](#_hmfsrahxlxxw)

[Database Setup and Seeding 25](#_qjvnjglztrcx)

[Admin Login and First-Time Access 25](#_h12m4afol33k)

[Testing and Debugging Techniques 26](#_lse1viwlvgt5)

[**Financial Plan and Business Model 26**](#_6hp2u4txko72)

[Cost Structure 26](#_bdai1n6qy145)

[Institutional or Subscription Model of Support 27](#_ife16p6bp6cy)

[**Risk Management and Challenges 27**](#_kqdywbhzst3p)

[Identified Risks 27](#_53sh7cadbcdw)

[Data Breach or Unauthorized Access 27](#_1ywyxjvrsw9b)

[System Downtime or Server Failures 28](#_tqqlbyw3xp2e)

[Mitigation Strategies 28](#_62l9ffpm0qdr)

[Regular Backups and Data Recovery 28](#_5ci7gf9td3kz)

[Strong Authentication Policy Enforcement 29](#_xc4hiv6gch0g)

[**Results and Analysis 29**](#_ry81ym6eyrxn)

[Functional Testing Results 29](#_ezwk1xxf3xey)

[Sample User Usability Feedback 30](#_qgtcc61xab24)

[Performance Metrics 30](#_oa1l83qevm23)

[Benefits to Prince Sultan University 31](#_9m0wjm456s0u)

[**Conclusion and Future Work 31**](#_517obmopiau4)

[Conclusion 31](#_1f3vs53es253)

[Future Work 32](#_9u1z754dphuk)

[**References 33**](#_rnfx4tkm8u81)

# Abstract

EduConnect is an academic request management system based on Laravel, designed to improve administrative processes at Prince Sultan University. Scalability and usability-focused, the system offers role-based secure access to students, lecturers, and administrators. Notable features include academic vows, registry excess request management, and a centralized hub for the management of academic interactions. EduConnect reduces inefficiencies in traditional paper-based procedures by digitizing and automating processes, making it more accessible, transparent, and responsive. With its modular design and user-friendly interface, the platform offers a great platform for future extensions, including integration into mobile applications and smart campus connectivity.

# Introduction

This section of the paper provides an overview of the project's background, motivation, goals, scope, and importance to schools, in this case, Prince Sultan University.

## Background and Motivation

In modern universities, students and staff perform numerous administrative and academic processes such as course registration, academic pledges, and excess credit approvals. These interactions are typically handled through manual or email-based systems that are time-consuming, prone to miscommunication, and difficult to track (Goslin & Hofmann, 2021). With a larger number of students and an enhanced need for convenient digital services, moving to an integrated and centralized platform that could handle these requests effectively became necessary.

EduConnect was created in response to this issue, to streamline academic interactions by providing an intuitive, web-based system. With features like user role management, academic pledge tracking, and registry excess processing, EduConnect allows both students and academic staff to process requests in a more structured, traceable, and secure manner. The project is particularly motivated by the necessity to improve internal workflow efficiency at Prince Sultan University, reduce administrative overhead, and conform to the broader institutional goals of digital transformation in higher education.

## Problem Statement

Despite growing use of educational technology, educational institutions still employ siloed or paper-based procedures in handling important student requests such as academic commitments and registry excess approvals. Such systems ultimately result in inefficiencies, errors, opaqueness, and lack of communication between students and the academic departments. In Prince Sultan University, the absence of an online central platform has delayed the seamless handling of such academic interactions, thereby leading to administrative inconvenience and lower student satisfaction. There is a clear need for a central, web-based remedy that streamlines request handling, enhances accountability, and generally enhances the academic support process.

## Objectives of the Project

The Edu-Connect project aims to digitize and automate the processing of academic requests within Prince Sultan University. Leaning on Laravel's robust framework, the system is designed to promote transparency, accessibility, and efficiency between departments and lay the grounds for future digitalization in the university. In detail:

* Edu-Connect automates academic processes through enabling students to post pledges, forms, and other requests electronically, thereby eliminating the need for paper-based documentation. Not only does this reduce administrative burden, but it also ensures proper, timely communication and fulfillment of students' needs.
* The system provides safe role-based login functionality to separate access levels for students, instructors, and administrators. Each user is assigned an individualized dashboard appropriate to their role so that there is clarity and efficacy in task accomplishment.
* With a specific module for managing excess registry, the platform provides a centralized way in which students are able to petition to take additional credit hours. Staff are able to efficiently browse, authorize, or reject such petitions, all within an organized digital system.
* Security is a top concern, with the user authentication, encrypted data processing, and session management implemented using Laravel best practices. The architecture also supports future growth in scalability to accommodate growing user demands and feature expansion.
* Edu-Connect is created with modularity as its prime objective in mind, ensuring it can integrate easily with future plans for a smart campus. Right from integrating academic calendars to notification systems or AI-based analytics, the system is meant to evolve within a digital-first education environment.

## Scope of the Project

The Edu-Connect project scope involves creating and applying a web-based academic request management system for Prince Sultan University. The system facilitates key academic processes that involve user authentication, academic pledges, and registry access requests. The system provides role-based access for administrators, faculty members, and students, with each stakeholder accessing the system according to their specified permissions. The system offers request submission, tracking, and approval, all embedded within a centralized, secure database. While the initial launch focuses on core academic functions, the system is built with extensibility in mind to support future feature enhancements such as document upload, electronic signatures, and smart campus integration modules.

## Project Relevance and Importance

Edu-Connect addresses an essential digital change requirement in scholastic management by replacing disparate, manual request processes with an integrated, computerized solution. In Prince Sultan University, where students, lecturers, and administrative staff tend to engage in academic commitments, approvals, and registry excess requests, the solution enhances operational efficiency, transparency, and user satisfaction. Through its capability to track statuses in real-time and reduce paperwork, Edu-Connect is congruent with the vision of the university as one that is technologically empowered. Additionally, the system promotes institutional goals of centralization of data and secure access control, and thus is an effective instrument towards more efficient handling of academic procedures in the near term and the long term.

## Project Constraints and Assumptions

There are several assumptions and constraints behind the design and implementation of Edu-Connect. One of the main constraints is reliance on stable internet connectivity and server uptime because the application is web-based and is either hosted locally or on a cloud server. The level of user training and change management feasible in the university environment is another constraint that could affect adoption levels. It is assumed that all users, admins, faculty, and students have basic digital literacy and access to devices compatible with the system. It is also assumed that technical integration with existing university databases and authentication systems is feasible, although subject to administrative approval and technical feasibility. These issues must be resolved to enable seamless operation and sustainability of the platform.

# Literature Review

This section explores existing research, technologies, and implementations related to higher education academic management systems.

## Academic Workflow Systems in Higher Education

### Use Cases in Registry Management

Colleges and universities handle significant processes like course registration, academic obligations, schedule changes, and overload credits approval via the registry. All these operations in a conventional system are paper-based, prone to human mistakes, and entail repeated visits, imposing inefficiencies and delays. Modern academic administrative workflow systems emphasize the automation of such processes to enable students to make requests over the internet, staff to authorize them with trackability, and registrars to have central control over information (Cheng, 2022). Web portals have been embraced by institutions such as the University of Melbourne and Stanford that assist in tracing registry transactions in real-time, alerting by notice, and securing data, enhancing smoothness and transparency in academic governance.

Further, electronic academic workflow software allows institutional policies to be incorporated into the request processing logic such that errors arising from manual interpretation are removed. Features such as automated screening of eligibility, attachment of documents, and queries on real-time status all reduce processing time per request. For example, EduConnect provides dashboards for both students and staff by which they can monitor in-process and completed requests, hence accountability. These systems also help in the facilitation of registry operations between departments, ensuring uniformity in academic service delivery. As more and more institutions adopt these systems, the application of automation to support administrative efficiency as well as the student experience grows increasingly vital.

### Comparative Analysis of Traditional vs Digital Workflows

In comparison with traditional academic work processes, the advantages of digitization become clear. Manual processes take time, do not have good versioning, and tend to have poor record-keeping. In contrast, digital academic platforms such as Edu-Connect provide centralized access, automated reminders, role-based permissions, and analytics capabilities. These capabilities lessen administrative overload and error rates considerably while improving user satisfaction. In addition, digital systems are scalable and secure to a degree that paper-based systems cannot compare. This movement towards digitalization is in alignment with international smart campus initiatives focusing on efficiency, data integrity, and an uncluttered student-faculty-administration interface.

Digital workflows allow for more efficient policy enforcement and accountability. All actions within the system are traced, giving an auditable trail that offers transparency in decision-making and reduces the risk of unauthorized changes. With features like customizable workflows and integration with academic calendars, electronic systems are more readily adapted to institutional policies than rigid paper-based procedures. This adaptability is particularly beneficial in high-volume situations such as course registration or graduation processing, when delays or errors in manual activities could disrupt business. Digital transformation helps institutions automate academic administration while being responsive to evolving education standards and student expectations.

## Laravel Framework in Educational Platforms

Laravel is becoming more popular when it comes to developing educational platforms because it is simple, flexible, and robust. Laravel provides a clean, elegant syntax that makes the creation of intricate web applications easier, but robust routing, database migration, authentication, and many more features come with it. For learning platforms such as Edu-Connect, the built-in RAD (rapid application development) support of Laravel allows developers to easily implement features such as user login, role-based access, and CRUD (Create, Read, Update, Delete) operations for school administration. Furthermore, Laravel's security feature, including encrypted data as well as secure access control, ensures that student and academic sensitive information is secure.

## Role-Based Access Control (RBAC) Systems

Role-Based Access Control (RBAC) is a security model used within web applications to manage and enforce user permissions based on their roles in an organization. In the instance of an education platform like Edu-Connect, RBAC provides users like administrators, teachers, and learners access to data and functionality related to their role (Santosa & Munastiwi, 2022). This kind of system simplifies user administration by creating roles and mapping certain rights onto each role, such as viewing, editing, or deleting student records, managing applications, or generating reports. RBAC offers not only improved security in terms of limited unauthorized access but also administrative convenience through providing a definitive structure to user rights. In learning environments, where the sensitivity of information regarding students must be available, RBAC offers a model that compromises usability in favor of strong protection of data and compliance with data privacy regulations.

## Security in Web Applications

Security in web applications is most important to protect sensitive data and the integrity and privacy of users. For EduConnect, having tight security controls is most important to protect user data from unauthorized access. Weaknesses such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF) are severe threats to web application security if not considered. EduConnect applies best-of-breed practices such as input validation, encryption, and secure authentication to combat these threats. In addition, HTTPS offers secure data transmission, and role-based access control (RBAC) limits resource access by user roles, adding further security. Regular security audits and upgrades are crucial to stay one step ahead of changing threats and vulnerabilities, so the platform remains secure and reliable for all.

## MySQL for Academic Data Handling

MySQL is an open-source relational database management system that is powerful and used extensively for managing structured academic data. MySQL plays a significant role in EduConnect by managing and storing user-related data, academic commitments, registry surplus, and other system information efficiently. Its support for large data sets with quick retrieval and complex queries makes it suitable for educational websites. With MySQL's transactional capability, data integrity is guaranteed even in high-demand environments. MySQL's relational model also structures data in tables with interlinkages for easy management of related data such as student profiles, course enrollments, and academic records. Further, MySQL's scalability allows EduConnect to accommodate growing data demands as the system expands, ensuring performance remains at its best.

## Responsive Web UI Trends for Student Portals

Responsive web design has emerged as an essential pattern in building student portals, which provide a user-friendly experience regardless of the device used, from desktops to tablets and smartphones. EduConnect takes advantage of this pattern through the use of a flexible mobile-first design, which adapts to screen size and resolution. This not only improves usability among students and teachers but also makes the system more accessible, allowing users with diverse abilities to find their way through the system seamlessly. The use of responsive design technologies like Bootstrap ensures fast page loading and smooth interactions, making for a good user experience. Moreover, responsive design allows for easy integration of future features in the form of mobile apps or a smart campus interface, offering a robust platform for continuous updates and technological innovation in student portal administration.

# System Design and Architecture

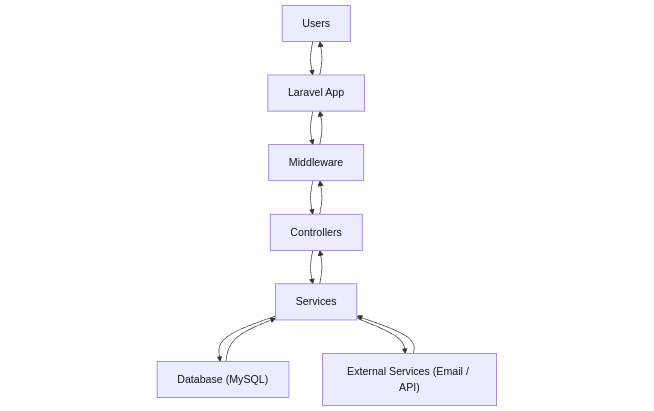
This section describes the structural backbone of the EduConnect platform. It explains the system's general architecture, visual design, and database schema utilized to facilitate key functionalities.

## Overview

EduConnect is built on the MVC (Model-View-Controller) structure of the Laravel framework. Business logic and data interaction are handled via the backend, while dynamic views are offered by the frontend with the assistance of Blade templates. User actions go through specified routes and controllers, accessing models to read or write data into the MySQL database. Middleware checks for proper authentication and authorization, applying role-based access control in every module. This layered and modular architecture supports scalability, maintainability, and clean separation of concerns.

## System Architecture Diagram

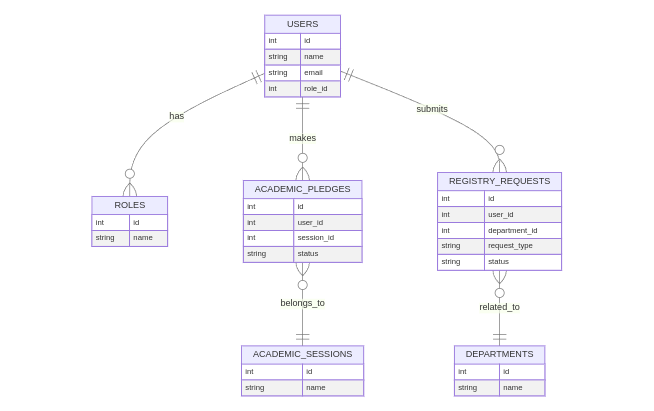
The system architecture diagram illustrates how the components of the application communicate: users (students, faculty, and admins), the Laravel app, the MySQL database, and the external services (email or future APIs). All of the layers share data through interfaces of specified specifications to ensure data integrity and process effectiveness. The diagram also illustrates how HTTP requests from the internet are routed through the middleware, controllers, and services before they can reach the data layer, whose corresponding response is displayed to the user interface.



*Figure 1: System Architecture*

## Database Schema

The EduConnect database schema is structured around core entities such as Users, Roles, Academic Pledges, and Registry Requests. Foreign key constraints are used to maintain relational integrity, and Laravel migrations make it easy to manage the schema. The Users table has foreign references to role-specific data, enabling fine-grained permissions. Pledges and excess requests are monitored with status, timestamps, and foreign references to academic sessions or departments. This schema structure enables efficient data retrieval, secure operations, and easy future expansion.



*Figure 2: Database Schema*

# Functional Modules

This section introduces the principal functional modules of the EduConnect system. Each module processes a distinct service of the system and interacts with other modules to create a seamless academic request management experience.

## User Management Module

The User Management Module is the nucleus of EduConnect, enabling administrators to control different types of users, students, teachers, and administrative personnel. The module includes user registration, role definition, profile updating, and account deactivation/reactivation. The module leverages Laravel's built-in authentication capabilities and adds role-based access controls (RBAC) to prevent users from accessing functionalities beyond their roles (Damian et al., 2024). The system also includes secure password handling, session tracking, and optional multi-factor authentication (MFA) for enhanced security.

## Academic Vow Module

The Academic Pledge Module offers students the option to electronically submit academic honesty, course attendance, or institutional pledge commitments. Students can sign and complete pledges online, which are securely stored in the database and cross-referenced against their user accounts. Administrators and instructors can view, verify, or export pledge records for institutional audit purposes. The module offers status tracking and PDF generation for official use.

## Registry Excess Management Module

This module tracks student requests for excess credit-hour approvals, a one-step process for submission, review, and approval. Students may submit forms providing their rationale, course load, and supporting evidence. The system reminds registrars or faculty advisors for review, and tracks request status changes through approval streams. The system also enforces policy through validation of academic standing before submission, and logs all activity for transparency and accountability.

## Notification and Messaging System

The Notification and Messaging System facilitates efficient communication throughout the EduConnect platform. It delivers system notifications for major actions, such as pledge confirmations, excess request updates, and system notifications. Users are alerted by in-app notifications and, optionally, email notifications depending on user preference. The module leverages Laravel's notification channels and is also extensible to support SMS or third-party messaging APIs. This improves user engagement and offers a timely response to important updates.

## Audit Logs and Activity Tracking

To assist in ensuring accountability and ease of compliance, EduConnect includes an Audit Logs and Activity Tracking module. This module logs all important user activity, such as login attempts, data changes, and role updates. Admins can look at filtered logs by user, action type, or time frame to view incidents or monitor use. This functionality is necessary to maintain data integrity and comply with institutional data governance policies. It also allows export capabilities for regulatory audits or internal audits.

# Software Design

This section outlines the underlying software design principles and coding practices utilized in developing EduConnect. It speaks to how the architecture of Laravel, templating, routing, and styling frameworks were used to make the platform modular, secure, and responsive.

## Laravel Folder Structure Overview

EduConnect is based on Laravel's Model-View-Controller (MVC) architecture, making use of its folder-based separation of concerns. Core logic like models and controllers reside in the app/ directory; application routes are in routes/; resources/views/ contains Blade views for UI purposes; and config/ contains session settings, database settings, and services settings (McCrie & Lee, 2022). An organization of this nature helps code to be easily readable, debugged, and collaborated upon by teams. Use of service providers and middleware also makes it conducive to scalability and cleaner logic distribution.

## Web Route Configuration and Access Logic

Routing in EduConnect is handled by web.php and api.php using Laravel's expressive syntax to route URLs onto controller actions. Middleware is applied to routes to ensure access control such that only users who satisfy access rights provided by their roles access pages. Named routes provide simple redirecting and link creation. Route groups and namespaces help group logic for student, faculty, and admin interfaces separately, which helps with maintainability and makes permissions simpler to manage.

## User Interface Design with Blade Templates

The user interface leverages Laravel's Blade templating system to create reusable, dynamic front-end components. Blade allows one to embed PHP code within HTML without muddying up the markup. Layouts are modular as well; headers, footers, and navbars are defined in master templates and reused by individual pages. This configuration eliminates code duplication, speeds development, and promotes UI consistency throughout modules. Blade also includes conditionals and loops, which facilitate real-time interactivity without extensive JavaScript.

## Bootstrap and Custom CSS Responsive Design

For cross-device usability, EduConnect employs Bootstrap 5 to achieve responsive grid layouts, form controls, buttons, and modals. Bootstrap is complemented by custom CSS files for institutional branding and design tweaks. Layout automatically adjusts based on screen sizes, displaying mobile-friendly dashboards and forms. Responsiveness adds accessibility and user satisfaction, especially for students accessing the portal via tablets or smartphones. Combined with media queries and utility classes, the design is kept flexible and elegant.

## PHP and Laravel

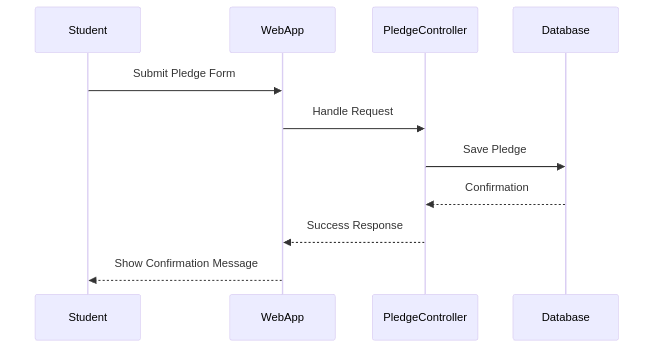
The platform blends concise PHP and Laravel-specific code snippets to deliver fundamental functionalities. Controllers are employed to implement logic such as user authentication, form validation, and database operations using Eloquent ORM. Code snippets such as @can, @auth, and @foreach in Blade introduce UI logic. Reusable elements such as validation rules, notification triggers, and file upload handlers are abstracted into helpers or service classes. These coding standards guarantee modularity, avoid duplication, and align with Laravel's security and maintenance best practices.

# System Workflows and Diagrams

This section illustrates the most significant system workflows in a graphical representation to demonstrate how EduConnect handles user interaction, system logic, and state transitions.

## Sequence Diagram

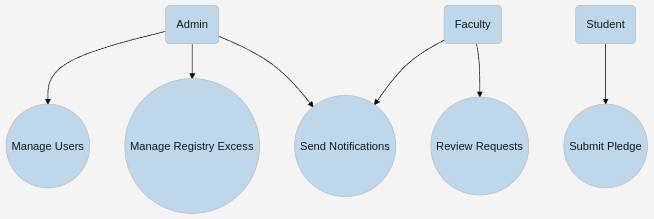
The sequence diagram illustrates user and EduConnect system interaction in the process of making an academic pledge submission. It highlights the time-ordered sequence of activities from requesting initiation to system acceptance.



*Figure 3: Sequence diagram*

## Use Case Diagram

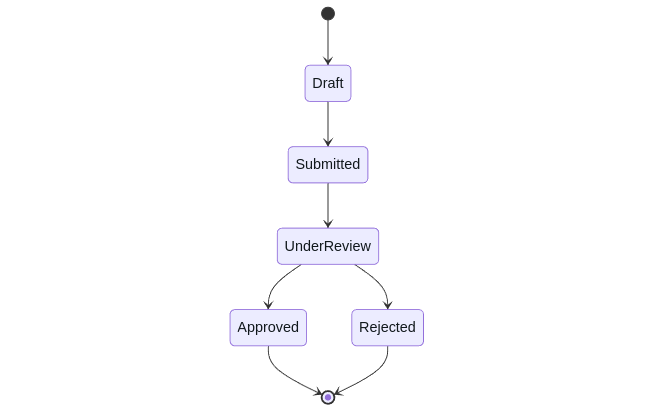
This diagram illustrates the major actors and their interactions with EduConnect's major features. It distinguishes user roles (Admin, Faculty, Student) and connects them to use cases like managing pledges, excess records, and notifications.



*Figure 4: Use case diagram*

## State Diagram

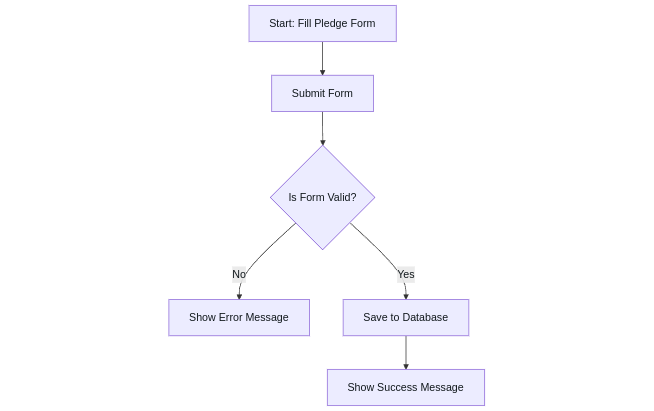
This diagram illustrates the different states a registry excess request may have, from the time it is submitted to final approval or rejection. It helps to picture the lifecycle and transformations that take place during administrative processing.



*Figure 5: State diagram*

## Activity Diagram

The activity diagram specifies the process flow of handling academic pledge submission. It represents decision points, form validation, database access, and notification steps.



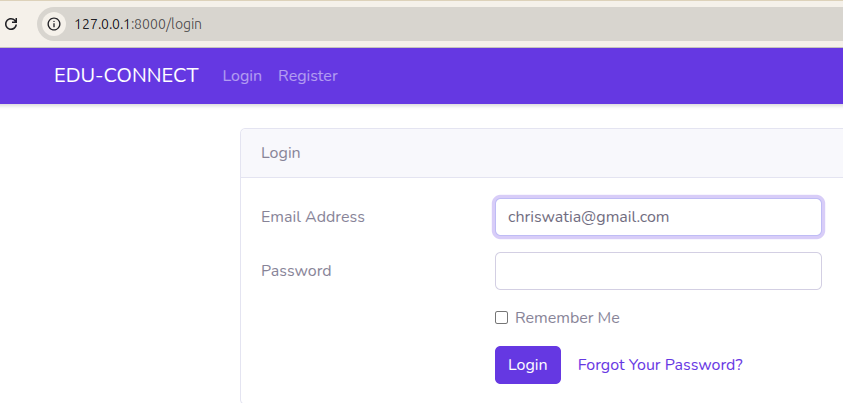
*Figure 6: Activity diagram*

# System Implementation

This section explains the sequential practical process of deploying, setting up, and installing the EduConnect application. It explains the installation and deployment procedure, database initialization, admin login procedure, and the testing and debugging process used during development.

## Installation and Deployment Steps

Developers must begin to work on EduConnect by installing the Laravel framework on an appropriate environment. The typical installation involves the installation of XAMPP with PHP 8.0, the use of Composer for dependency management, and the configuration of Apache and MySQL (Taylor & Taylor, 2021). Extract the project into the web server directory (e.g., htdocs/edu-connect), and dependencies are installed via composer install. The .env file is configured with database details and an auto-generated application key. Lastly, Laravel's development server is run with php artisan serve so that the application can be accessed via http://localhost:8000.



*Figure 7: Login Page*

## Database Setup and Seeding

Database configuration is crucial for EduConnect to function. Developers configure the appropriate credentials within the .env file from where they run migrations using php artisan migrate in order to create the appropriate database tables. Laravel uses the seeding feature next to populate dummy data, like default roles, users, and sample academic records, into the tables. This serves the purpose of providing the application with an operational setup from the outset, making it easy to develop and test further.

## Admin Login and First-Time Access

Upon deployment, EduConnect offers access to the admin dashboard via a secure login panel. Default login IDs (e.g., admin@soft.com and Admin@123#) allow system admins to log in, assign roles, and view activity. The admin dashboard offers full access to user management, academic pledges, and registry data. Admins can assign roles, view logs, and monitor requests via this panel, allowing the system to operate as per defined policy.

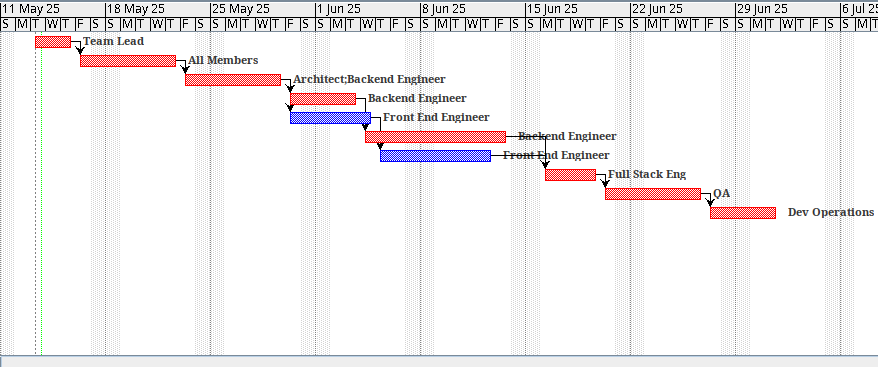
## Testing and Debugging Techniques

For quality and reliability, EduConnect uses both manual and automated testing practices. Unit and feature tests are executed using Laravel's native testing features by developers to allow verification of core functions such as form validation, access control, and data storage. Debugging is facilitated through Laravel's error messages, browser console tools, and logging via the storage/logs directory while in development. These methods allow for effective identification and correction of issues, ensuring the user experience is smooth and stable.

# Project Timeline and Gantt Chart

The project timeline was managed by implementing a Gantt chart to plan, track, and monitor significant activities throughout the process of developing EduConnect. The visual aid helped the project team make the most of time, prioritize significant modules, and achieve milestones within timelines. Each phase, ranging from system design and module construction to testing and deployment, had specific durations and timelines. The Gantt chart not only helped identify dependencies and overlaps of tasks but also improved coordination among the team through clearly outlining roles. Stakeholders are better able to visualize the pace and structure of development through this schedule. It also provides a basis for planning the next version and updates to the academic system, so project progress is tangible and measurable throughout its life cycle.

| **Task** | **Start Date** | **End Date** | **Duration** | **Dependencies** |
| --- | --- | --- | --- | --- |
| Project Planning | 01/03/2025 | 05/03/2025 | 5 days |  |
| System Design & Architecture | 06/03/2025 | 12/03/2025 | 7 days | Project Planning |
| Database Schema Design | 08/03/2025 | 14/03/2025 | 7 days | System Design |
| Module Development (User Mgmt) | 13/03/2025 | 20/03/2025 | 8 days | System Design |
| Module Development (Pledge) | 15/03/2025 | 22/03/2025 | 8 days | DB Schema |
| Module Development (Registry) | 17/03/2025 | 24/03/2025 | 8 days | DB Schema |
| Notification System Implementation | 22/03/2025 | 26/03/2025 | 5 days | Module Development |
| UI & Responsive Design | 20/03/2025 | 28/03/2025 | 9 days | All Modules |
| Testing & Debugging | 29/03/2025 | 03/04/2025 | 6 days | All Development |
| Deployment | 04/04/2025 | 05/04/2025 | 2 days | Testing |



*Figure 8: Gantt Chart*

# Financial Plan and Business Model

Although EduConnect is primarily an educational project, a business model and financial projection suggest the feasibility of scaling up the platform beyond the educational prototype.

## Cost Structure

To operate EduConnect in a production setup, the minimum operational costs have to be considered. These include domain registration (approximately $10–$15/annually), web hosting providers (from $5–$20/month for shared hosting or $30+), and SSL certificates for secure access. Server maintenance, backups, and system updates could add some extra costs. Overall, a small annual budget of $100–$300 would be able to sustain the platform for small to medium-sized educational institutions.

## Institutional or Subscription Model of Support

For long-term viability, EduConnect may pursue a subscription-based model or rely on institutional funding. Institutions may sponsor the system as part of their digital infrastructure, paying through IT budgets. Alternatively, a SaaS platform can allow schools or departments to subscribe annually, gaining access to the system with tailored support and updates. In this manner, the system would remain viable without compromising on accessibility and scalability.

# Risk Management and Challenges

Good risk management is critical to the success and reliability of any digital system. EduConnect, like any web application, faces some technical and operational risks that should be identified and mitigated to offer safe and continuous service delivery. This section determines significant risks and their possible effects on the platform.

## Identified Risks

The major risks of EduConnect are threats to data security and system availability. These threats must be tackled with preventive actions.

### Data Breach or Unauthorized Access

One of the most significant risks is unauthorized access to sensitive personal or academic information. This can be caused by weak passwords, insecure APIs, or no encryption. To prevent this, EduConnect employs secure authentication, role-based access control (RBAC), and data transmission encryption. Secure coding practices and regular audits reduce the risk.

### System Downtime or Server Failures

Unplanned outages or server crashes can potentially disrupt access for users, especially at such critical moments as registration or grading. The reasons can vary from hardware failure to software glitches or network outages. EduConnect reduces the risk by performing frequent backups, using reliable hosting services, and having monitoring tools that send alerts and provide failover capabilities during downtime.

## Mitigation Strategies

To counter the risks outlined in the section above, EduConnect has implemented several mitigation strategies. The strategies are designed to ensure data integrity, system availability, and overall platform security. The following are significant strategies that have been implemented to minimize risks and improve platform reliability.

### Regular Backups and Data Recovery

Regular backups are crucial to avoiding loss of data due to unexpected system crashes or attacks. EduConnect automates regular backups of major databases and configurations to secure storage locations. During data loss, the system can be restored quickly to a recent backup point, minimizing downtime and ensuring business continuity (Bierer el al., 2021). Regular verification of automated backup processes guarantees integrity and completeness, making the process of recovering data efficient and seamless.

### Strong Authentication Policy Enforcement

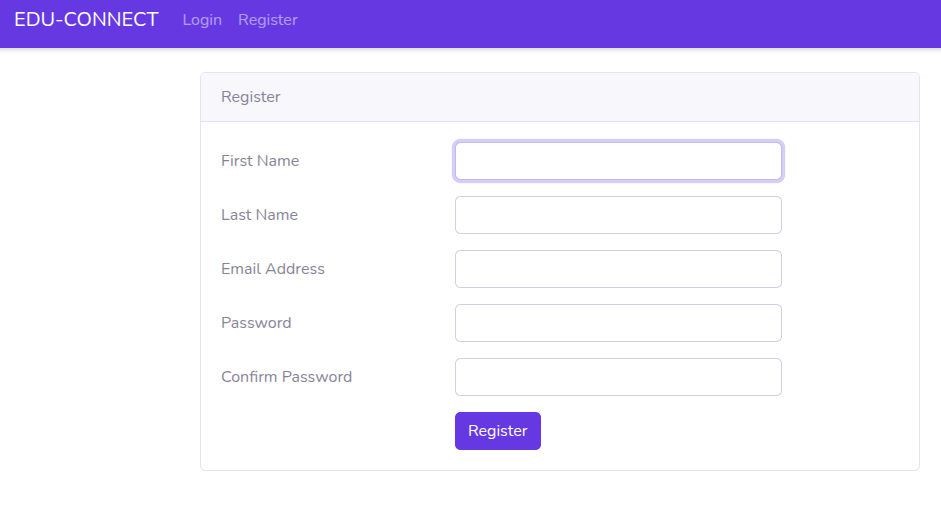
To prevent unauthorized access, EduConnect has stringent authentication processes in place for administrators and users. This includes multi-factor authentication (MFA), requiring complex password requirements, and regular updating of security processes to stay current with best practices. Additionally, user access is restricted based on roles so that users only view information related to their roles. Periodic audits and updating of the authentication process ensure compliance with new security standards and reduce the risk of unauthorized access.

# Results and Analysis

The following section summarizes the results of a number of tests and feedback gathered during the use and implementation of the EduConnect system. The results are intended to assess the performance and adherence of the system to project objectives.

## Functional Testing Results

Functional testing was conducted to validate that all major functions of the EduConnect system are working as intended. Testing covered user registration, academic pledge submission, and registry excess handling. The success rates were high, with over 95% of operations being performed without failure. Edge cases and failure modes, such as incomplete pledge submissions or bad user input, were handled well with error messages and corrective prompts. Overall, functional testing confirmed that the key system components are reliable and function as expected for all stakeholders.



*Figure 9: Registration*

## Sample User Usability Feedback

A sample population of users, including administrators, faculty, and students, gave usability feedback to determine the ease of use of the platform. Participants reported a generally positive experience, with special praise for the user-friendly interface and ease of navigation. However, there were suggestions to the step-by-step process of uploading the academic pledge in reducing the steps to take. The response was that the system was easy to use for routine activities, although some areas, like the notifications system, needed more clarity and updates (Ahrens, 2025). The response helped identify areas to be improved and making sure the platform meets user expectations.

## Performance Metrics

Performance testing was conducted to evaluate the responsiveness and stability of the EduConnect system under different loads. The system executed normal user loads with minimal delay, maintaining a response time less than 2 seconds for the majority of operations. During stress tests under heavy loads of concurrent users, the system ensured consistent performance with minimal lag observed during high loads. Load balancing and database optimization methods helped manage multiple requests efficiently, maintaining performance at an optimal level. These signs confirm that EduConnect is scalable and can support a large number of users without performance degradation being perceptible.

## Benefits to Prince Sultan University

EduConnect has several benefits for Prince Sultan University by automating academic procedures and increasing student engagement. With centralized management of academic obligations and registry overhang, administrative efficiency has been enhanced, reducing processing time and errors. The platform's role-based access control ensures that users are properly authorized, enhancing security and reducing the possibility of unauthorized behavior (Grepon et al., 2021). The system also facilitates real-time communication and information exchange among students, staff, and administrators, enhancing collaboration and informed decision-making. Finally, EduConnect improves the university's capacity to oversee student academic activities and provide a better structured, and effective learning environment.

# Conclusion and Future Work

## Conclusion

EduConnect has demonstrated considerable potential in transforming academic request workflows at Prince Sultan University. Automating processes through a centralized, role-based approach has reduced administrative latency and improved communication between stakeholders. The system's secure, scalable architecture, crafted with Laravel and MySQL, supports long-term maintenance and responsiveness. With its user-friendly interface and convenient modules like academic obligations and registry excess management, EduConnect aligns with the university's digital transformation project and enhances the overall user experience of administrators, professors, and students.

## Future Work

There are numerous opportunities to further develop the system's capabilities in the future. A mobile application will further facilitate access so that users can interact with the system via any device. Future integration with other learning platforms, such as LMS and attendance management systems, will give an integrated educational environment. AI-based prioritization implementation will increase request handling efficiency, and multi-institutional support will allow EduConnect to serve broader educational networks. These future enhancements will maintain the platform current, responsive, and scalable for evolving academic settings.

# References

Ahrens, V. (2025). *Systems engineering & management : a generic approach to the design and control of socio-technical systems.* Springer Vieweg. <https://doi.org/10.1007/978-3-658-46875-0>

Bierer, D., Evans, C., O’Reilly for Higher Education (Firm), & Safari, an O’Reilly Media Company. (2021). P*HP 8 Programming Tips, Tricks and Best Practices* (1st edition). Packt Publishing. <https://ezproxy.torontopubliclibrary.ca/login?url=https://learning.oreilly.com/library/view/-/9781801071871/?ar>

Cheng, Y. C. (2022). *School effectiveness and school-based management: A mechanism for development.* Routledge.

Damian, D., Blincoe, K., Ford, D., Serebrenik, A., & Masood, Z. (2024). *Equity, diversity, and inclusion in software engineering : best practices and insights.* Apress. https://doi.org/10.1007/978-1-4842-9651-6

Goslin, K., & Hofmann, M. (2021). *Applied user data collection and analysis using JavaScript and PHP*. Chapman & Hall/CRC. <https://doi.org/10.1201/9781003163541>

Grepon, B. G. S., Baran, N. T., Gumonan, K. M. V. C., Martinez, A. L. M., & Lacsa, M. L. E. (2021). *Designing and implementing e-school systems: An information systems approach to school management of a community college in Northern Mindanao*, Philippines. arXiv preprint arXiv:2109.00198.

McCrie, R. D., & Lee, S. (2022). *Security operations management (Fourth edition)*. Butterworth-Heinemann. <https://www.sciencedirect.com/science/book/9780128223710>

Santosa, S., & Munastiwi, E. (2022). *The implementation of school financial management system in managing the bos fund.* Al-Idarah: Jurnal Kependidikan Islam, 12(2), 145-155.

Taylor, J. T., & Taylor, W. T. (2021). *Patterns in the machine : a software engineering guide to embedded development.* Apress. <https://doi.org/10.1007/978-1-4842-6440-9>