**Mariam Srour**

**INTRODUCTION**

1. **Purpose:**

The purpose of this document is to establish an online system that efficiently manages enrollment and passenger information, aiming to simplify the application and enrollment process.

1. **Intended Audience and Reading Suggestions**

This project serves as a pioneering prototype for the student enrollment management system. It has been skillfully developed under the expert guidance of esteemed college professors. This ingenious system plays a pivotal role in streamlining the applications and enrollment processes for new students, offering a comprehensive solution for managing applications, diligently tracking their status, and facilitating seamless communication with prospective students. The project's significance extends not only to the enrollment team but also to the aspiring students who benefit from a smoother and more efficient application experience.

1. **Project Scope**

The primary purpose of this web-based project is to establish a seamless and efficient application and enrollment process for new students interested in joining the institution. By developing a digital platform, the project aims to modernize and simplify the traditionally paper-intensive and time-consuming application procedure. The goal is to create a more accessible, user-friendly, and technologically advanced system that attracts prospective students and provides them with a positive first impression of the institution.

1. **References**

<https://krazytech.com/projects>

**Overall Description**

1. **Product Perspective**

The proposed web-based project, with a focus on student enrollment, encompasses the following key components:

* Student Information:The system will capture and store essential details of prospective students, such as their unique student code(new\_student\_id), full name, address, and contact number. This information will serve as a comprehensive record to facilitate smooth communication and efficient handling of student-related matters.
* Application:The application module is designed to record and manage the application details of each prospective student. It encompasses the following key information: Application ID, Student ID, Date of Application, Preferred Course/Program, Supporting Documents, and Application Status

1. **Product Features**

The major features of the enrollment database system as shown in below entity–relationship model (ER model):

1. **User Class and Characteristics**

The proposed web-based platform aims to provide a streamlined and user-friendly experience for new students seeking admission to the university. Through this website, prospective students can efficiently submit their applications, enabling them to gain a comprehensive understanding of the university's offerings. Simultaneously, the system serves as a centralized repository for all new student applications, allowing university officials to effectively review, accept, or reject applicants while facilitating seamless communication.

Key features of the application include:

* Application Submission and Assessment: Prospective students can easily submit their applications through the website. Upon evaluation of their applications and any necessary exams, the system provides prompt feedback, indicating whether the student meets the admission criteria immediately or requires remedial support.
* Personalized Remedial Recommendations: For students who may benefit from additional support, the application identifies the specific areas requiring improvement. It offers personalized remedial recommendations to help the student enhance their knowledge and skills to meet the university's standards.
* Real-time Communication: The web-based application enables real-time communication between the university and the student. It serves as a platform for notifications, updates, and clarifications, ensuring effective and efficient information exchange throughout the admission process.
* Transparent Assessment Results: After completing any necessary exams or assessments, the application provides clear and transparent evaluation results. Students can access their scores and feedback promptly, empowering them to understand their performance and progress.
* AI-driven Decision-making: The application utilizes artificial intelligence algorithms to enhance the decision-making process. It can identify potential students who meet the admission criteria immediately, based on their application and exam performance, thus expediting the acceptance process.
* Seamless Student Records: The application maintains a comprehensive student database, storing essential information, exam results, and remedial progress. This seamless record-keeping system ensures that all relevant data is readily accessible to university officials and students as needed.
* User-Friendly Interface: The user-friendly interface of the application fosters a positive experience for students, making it easy to navigate through the various functionalities. It empowers students to actively engage with the application and monitor their admission journey.

**requirements are more specific and the requirements parts**

1. **Operating Environment**

* Client/server system: a relationship in which one program, the client, requests a service or resource from another program, the server.
* Operating system: Microsoft Windows, macOS, and Linux since .NET Core, designed for cross-platform compatibility, operates seamlessly on Microsoft Windows, macOS, and Linux.
* Database: mysql database
* Platform: .NET Core/ css/ html/ js/php…

**SYSTEM FEATURES**

1. **DESCRIPTION and PRIORITY**

The web-based enrollment project holds high priority as it streamlines the application process for new college students, maintaining comprehensive records of submitted applications, personal preferences, payment procedures, and admission statuses. With centralized information accessible through the website, students can easily understand what's required and learn more about the college without the need for direct communication. The project's significance lies in its ability to enhance the student experience, optimize application management for university staff, and aid in proactive planning for accommodating increasing student numbers each year.

1. **FUNCTIONAL REQUIREMENTS**

* User Registration: Allow new students to register and create unique accounts on the web platform.
* Application Submission: Provide an interface for students to submit their enrollment applications online.
* Document Upload: Allow students to upload necessary documents, such as academic records and identification.
* Application Tracking: Enable students to track the status of their applications in real-time.
* Payment Tracking: Allow students to monitor their payment progress by providing a transparent view of the total amount paid, remaining balance, and details of previous transactions, ensuring clarity and accountability in the payment process.
* Admission Status: Notify students about their admission status (accepted, rejected, or pending) through the platform.
* Remedial Recommendations: Provide personalized remedial suggestions for students who require additional support.
* Course Selection: Allow accepted students to select their preferred courses or programs.
* Communication Platform: Offer a messaging system or another way that facilitate communication between students and university officials.
* Course Catalog: Provide an updated catalog of available courses and programs with detailed descriptions.
* Student Profile: Allow students to update their personal information and preferences.
* Enrollment Confirmation: Enable accepted students to confirm their enrollment and secure their spot.
* Student Dashboard: Offer a personalized dashboard for each student to view relevant information at a glance.
* Application Deadline Alerts: Send automatic notifications to remind students of approaching application deadlines.
* Acceptance Notifications: Notify students about acceptance decisions promptly via email or in-app notifications.
* Withdrawal Process: Allow students to withdraw their applications or enrollment if needed.
* Multi-platform Support: Make the application compatible with various devices and operating systems.
* Security Measures: Implement robust security measures to safeguard student data and payment information.
* Data Backup: Regularly backup application data to prevent loss in case of system failures.
* User Support: Provide a help center or support system for students to address any technical or application-related issues.
* Analytics and Reporting: Generate reports and analytics on application trends and enrollment statistics for university administrators.
* System Performance: Ensure the platform maintains high performance and responsiveness even during peak times of application submission

## EXTERNAL INTERFACE REQUIREMENTS

1. **USER INTERFACES**

* Front-end software: .net core version 6
* Back-end software: MySQL

1. **HARDWARE INTERFACES**

* Microsoft Windows, macOS, and Linux
* A browser that supports HTML & Javascript & CSS.

1. **SOFTWARE INTERFACES**

Operating system: The chosen operating system is Microsoft Windows, macOS, and Linux due to .NET Core's cross-platform compatibility, ensuring seamless operation across different systems.

Database: The project uses the MySQL database to store students’ records and other important records efficiently.

Platform: The project utilizes a combination of technologies, including .NET Core, CSS, HTML, JS, and PHP, to create a robust and dynamic web application.

## NONFUNCTIONAL REQUIREMENTS

## PERFORMANCE REQUIREMENTS

## ER diagram: The E-R Diagram constitutes a technique for representing the logical structure of a database in a pictorial manner. This analysis is then used to organize data as a relation, normalizing relation, and finally obtaining a relation database.

**ENTITIES:**These specify distinct real-world items in an application.

**PROPERTIES/ATTRIBUTES:** These specify properties of an entity and relationships. “ I did not include attributes in the diagram above since there are numerous attributes to consider. However, the attributes for the entities depicted in the diagram can be found listed under the respective entities in the diagram below.”

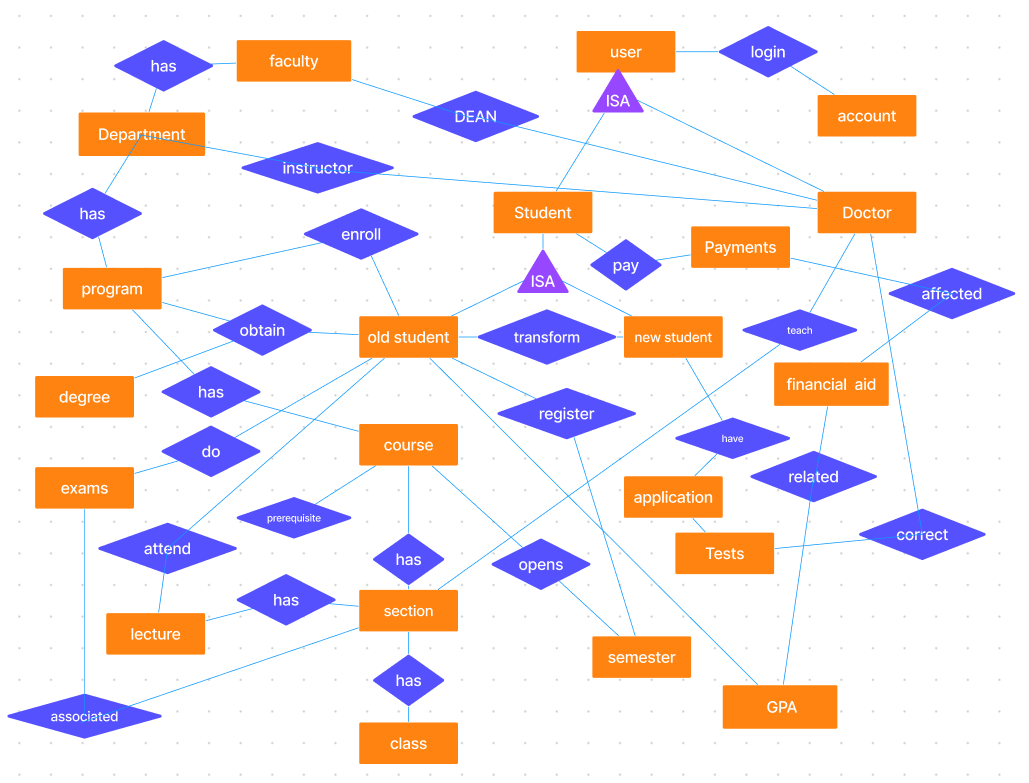
**RELATIONSHIPS:** These connect entities and represent meaningful dependencies between them.

## “This ER diagram is for the SIS not for new student only”

## Link of the ER diagram:

## <https://www.figma.com/file/y7dKyDJWWRe6VM6fZ6B5AS/Untitled?type=whiteboard&node-id=0%3A1&t=8cAyXRoeDkPq55Fr-1>

## Diagram:



## Database diagram:

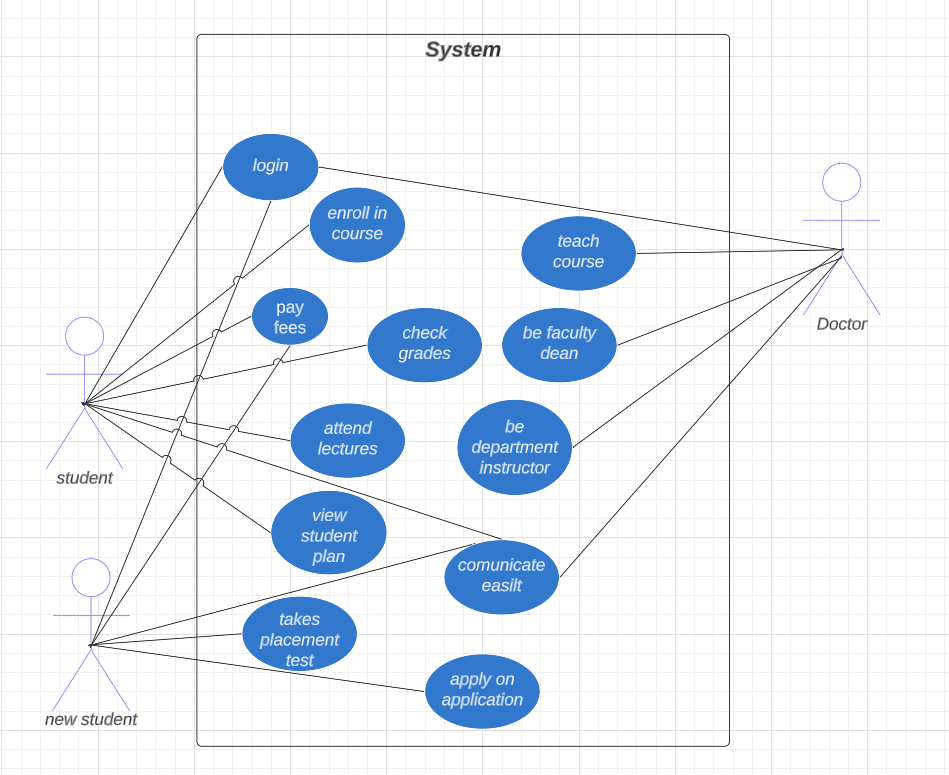
## The Database Diagram is a visual representation of a database's logical structure, facilitating analysis, entity-relationship organization, and normalization for a well-structured relational database. It enhances data management, promotes data integrity, and streamlines data operations.

## Link: https://dbdiagram.io/d/64ca707e02bd1c4a5e213fc2



## Use case diagram:

## The Use Case Diagram is a graphical depiction of the functional requirements of a system, illustrating interactions between actors and use cases. It aids in understanding system behavior, identifying user interactions, and guiding system development. Use Case Diagrams simplify communication between stakeholders and assist in defining the scope of the software system.



## Normalization:

## Normalization is a critical database design objective that aims to minimize redundancy and enhance data integrity. By storing information only once, it optimizes storage space and reduces data duplication, resulting in a more efficient and organized database. Through the process of breaking down tables into smaller, theme-specific ones, normalization eliminates modification anomalies and data inconsistencies that may arise from improper database design. It encompasses three primary normal forms: first normal form (1NF), second normal form (2NF), and third normal form (3NF), which ensure proper data organization and dependency management. However, normalization should be applied with careful analysis and a comprehensive understanding of its implications to achieve the desired benefits of improved data consistency and efficient query performance, making it suitable for most practical applications.

## SAFETY REQUIREMENTS

## For safety requirements, the system must have robust mechanisms in place to handle catastrophic failures that may cause extensive damage to the database, such as disk crashes. In such scenarios, a reliable recovery method should be implemented. This method involves restoring a previously backed-up copy of the database, which is typically stored in archival storage, such as tape. To ensure data integrity and accuracy, the system then reconstructs the most current state of the database by reapplying or redoing the operations of committed transactions from the backed-up log, up to the time of failure. This ensures that critical data is not lost and that the database can be restored to a consistent state after a catastrophic event.

## SECURITY REQUIREMENTS

## In our project, security systems rely on efficient and reliable database storage, much like many other applications. However, the unique demands of the security market emphasize the importance of carefully selecting the appropriate database partner. Given the critical nature of security systems, vendors must prioritize factors such as data security, scalability, real-time data processing, and high availability. Choosing the right database partner is crucial to ensure seamless integration, robust performance, and adherence to strict security standards, ultimately providing a trustworthy and effective solution for our security-focused clientele.

## SOFTWARE QUALITY ATTRIBUTES

## 1. Availability: The university management system should be consistently available to accommodate various academic activities and administrative tasks. With a large number of students, faculty, and staff relying on the system, it is essential to maintain continuous availability.

## 2. Correctness: Accuracy and correctness are paramount in the university management system. The software must handle student enrollment, course registration, grading, and other administrative processes accurately to ensure data integrity and avoid errors.

## 3. Maintainability: To support ongoing improvements and adaptability to changing requirements, the web application must be highly maintainable. Regular updates, bug fixes, and enhancements are necessary to keep the system efficient and compatible with evolving technologies.

## 4. Usability: User-friendliness is crucial for the university management system. Students, faculty, and administrators should find it intuitive and easy to navigate. The system should cater to various user needs, allowing access from different devices to enhance usability.